

PRZEDMIOTOWE OCENIANIE UCZNIA – INFORMATYKA

I Liceum Ogólnokształcące im. Juliusza Słowackiego w Skarżysku-Kamiennej

rok szkolny 2026/2027

zakres podstawowy i rozszerzony

*Dokument opracowany z uwzględnieniem Statutu Szkoły oraz planów realizacji informatyki
w klasach I–IV.*

Spis treści

POSTANOWIENIA OGÓLNE	6
1. Wymagania edukacyjne na poszczególne stopnie szkolne	6
2. Formy i obszary sprawdzania osiągnięć edukacyjnych	7
2.1. Formy sprawdzania	7
2.2. Obszary sprawdzania	8
2.3. Organizacja sprawdzania	8
2.4. Skala procentowa sprawdzianów pisemnych	9
2.5. Minimalna liczba ocen	9
2.6. Samodzielność i korzystanie ze źródeł	9
3. Kryteria oceniania	9
3.1. Zasady ogólne	9
3.2. Zadania praktyczne	10
3.3. Zadania programistyczne i algorytmiczne	10
3.4. Projekty	11
3.5. Odpowiedzi ustne i wyjaśnienie rozwiązania	11
3.6. Autorstwo i narzędzia AI	11
3.7. Ocena klasyfikacyjna	11
4. Narzędzia i zasady poprawiania ocen	12
4.1. Zasady ogólne	12
4.2. Prawo do poprawy ocen bieżących	12
4.3. Termin poprawy	12
4.4. Poprawa różnych form	12
4.5. Nieobecność i zaległe prace	13
4.6. Nieprzygotowanie	13
4.7. Wynik poprawy	13
5. Dostosowanie wymagań edukacyjnych	13
5.1. Zasady ogólne	13
5.2. Dysleksja	13
5.3. Dysgrafia	14
5.4. Dyskalkulia	14
5.5. Spektrum autyzmu	14

5.6. Trudności w koncentracji uwagi, w tym ADHD.....	14
5.7. Ograniczenia wzrokowe.....	15
5.8. Ograniczenia słuchowe.....	15
5.9. Ograniczenia sprawności ruchowej	15
5.10. Trudności językowe	15
5.11. Uczeń szczególnie uzdolniony	15
5.12. Granice i indywidualizacja dostosowania.....	16
6. Warunki i tryb uzyskiwania rocznej oceny klasyfikacyjnej wyższej niż przewidywana (proponowana)	16
6.1. Zasady i warunki	16
6.2. Zgłoszenie i termin	16
6.3. Forma i zakres sprawdzianu rocznego.....	17
6.4. Ocenianie, samodzielność i dostosowania	17
6.5. Wynik	17
7. Zasady informowania o ocenach.....	17
7.1. Informowanie o zasadach oceniania.....	17
7.2. Informowanie o ocenach bieżących.....	18
7.3. Informowanie rodziców	18
7.4. Oceny przewidywane i proponowane	18
7.5. Jawność i czytelność oceniania	18
ZBIORCZE WYMAGANIA EDUKACYJNE NA POSZCZEGÓLNE OCENY.....	19
KLASA I – zakres podstawowy.....	19
Ocena dopuszczająca – 2.....	19
Ocena dostateczna – 3.....	19
Ocena dobra – 4	20
Ocena bardzo dobra – 5	20
Ocena celująca – 6	20
Ocena niedostateczna – 1	20
KLASA I – zakres rozszerzony	22
Ocena dopuszczająca – 2.....	22
Ocena dostateczna – 3.....	22
Ocena dobra – 4	22

Ocena bardzo dobra – 5	22
Ocena celująca – 6	23
Ocena niedostateczna – 1	23
KLASA II – zakres podstawowy	24
Ocena dopuszczająca – 2	24
Ocena dostateczna – 3	24
Ocena dobra – 4	24
Ocena bardzo dobra – 5	24
Ocena celująca – 6	25
Ocena niedostateczna – 1	25
KLASA II – zakres rozszerzony	26
Ocena dopuszczająca – 2	26
Ocena dostateczna – 3	26
Ocena dobra – 4	26
Ocena bardzo dobra – 5	26
Ocena celująca – 6	27
Ocena niedostateczna – 1	27
KLASA III – zakres podstawowy	28
Ocena dopuszczająca – 2	28
Ocena dostateczna – 3	28
Ocena dobra – 4	28
Ocena bardzo dobra – 5	28
Ocena celująca – 6	29
Ocena niedostateczna – 1	29
KLASA III – zakres rozszerzony	30
Ocena dopuszczająca – 2	30
Ocena dostateczna – 3	30
Ocena dobra – 4	30
Ocena bardzo dobra – 5	30
Ocena celująca – 6	31
Ocena niedostateczna – 1	31

KLASA IV – zakres rozszerzony	32
Ocena dopuszczająca – 2.....	32
Ocena dostateczna – 3.....	32
Ocena dobra – 4	32
Ocena bardzo dobra – 5	32
Ocena celująca – 6	33
Ocena niedostateczna – 1	33
POSTANOWIENIA KOŃCOWE	34
PROPOZYCJA TEMATYKI ZAJĘĆ Z INFORMATYKI.....	35
Założenia dokumentu.....	35
KLASA I – ZAKRES PODSTAWOWY	36
KLASA I – ZAKRES ROZSZERZONY.....	37
KLASA II – ZAKRES PODSTAWOWY	39
KLASA II – ZAKRES ROZSZERZONY	41
KLASA III – ZAKRES PODSTAWOWY	43
KLASA III – ZAKRES ROZSZERZONY	44
KLASA IV – ZAKRES ROZSZERZONY.....	46
Propozycje organizacji pracy projektowej	48
Uwagi dotyczące doboru narzędzi	48

POSTANOWIENIA OGÓLNE

1. Przedmiotowe Ocenianie Ucznia (POU) określa zasady oceniania osiągnięć edukacyjnych z informatyki w klasach I–IV.
2. POU obejmuje zakres podstawowy i rozszerzony odpowiednio do realizowanych w danej klasie planów nauczania.
3. Ocenianie odbywa się w odniesieniu do wymagań edukacyjnych wynikających z realizowanego programu i planu pracy nauczyciela.
4. W zakresie rozszerzonym uczeń spełnia wymagania właściwe dla zakresu podstawowego, a ponadto wymagania dodatkowe określone dla zakresu rozszerzonego.
5. Szczegółowe wymagania edukacyjne dla poszczególnych klas i zakresów stanowią integralne załączniki do niniejszego POU.
6. Treści realizowane w klasie IV, które w historycznym planie pracy były wyodrębnione jako zakres eksperymentalny, w niniejszym POU traktowane są jako część realizowanego zakresu rozszerzonego.
7. W sprawach nieuregulowanych w niniejszym POU stosuje się postanowienia Statutu Szkoły oraz obowiązujące przepisy prawa oświatowego. W przypadku zmiany postanowień Statutu mających wpływ na ocenianie stosuje się aktualne postanowienia Statutu.

1. Wymagania edukacyjne na poszczególne stopnie szkolne

Wymagania edukacyjne określają wiadomości i umiejętności, których opanowanie jest niezbędne do uzyskania poszczególnych ocen klasyfikacyjnych z informatyki. Wymagania mają charakter kumulatywny – wyższy poziom oceny obejmuje wymagania przewidziane dla ocen niższych.

W informatyce szczególne znaczenie ma praktyczne wykorzystanie wiadomości: dobór narzędzi, samodzielne rozwiązywanie problemów, tworzenie i testowanie rozwiązań, analiza błędów oraz umiejętność wyjaśnienia sposobu postępowania.

Ocena	Wymagania edukacyjne
6 – celujący	Uczeń w pełni opanował wymagane wiadomości i umiejętności. Samodzielnie i twórczo rozwiązuje złożone lub nietypowe problemy, integruje wiadomości z różnych obszarów, świadomie dobiera metody i narzędzia, analizuje poprawność i efektywność rozwiązań, uzasadnia wybór oraz potrafi rozwiązywanie rozwijać i optymalizować.

Ocena	Wymagania edukacyjne
5 – bardzo dobry	Uczeń bardzo dobrze opanował wymagane wiadomości i umiejętności. Samodzielnie rozwiązuje złożone problemy, świadomie dobiera metody i narzędzia, analizuje warianty, uzasadnia sposób postępowania i stosuje poznane umiejętności w nowych sytuacjach.
4 – dobry	Uczeń dobrze opanował wymagane wiadomości i umiejętności. Samodzielnie rozwiązuje typowe problemy, poprawnie dobiera spośród poznanych metod, dostosowuje znany sposób postępowania do częściowo nowej sytuacji oraz wykrywa i poprawia typowe błędy.
3 – dostateczny	Uczeń opanował podstawowe wiadomości i umiejętności. Samodzielnie wykonuje typowe zadania o podstawowym stopniu trudności, rozumie zasadniczy sens stosowanych metod i dokonuje podstawowej kontroli otrzymanego rezultatu.
2 – dopuszczający	Uczeń opanował elementarne wiadomości i umiejętności konieczne do dalszego kształcenia. Wykonuje proste zadania, często korzystając ze wzoru, instrukcji lub pomocy nauczyciela.
1 – niedostateczny	Uczeń nie spełnia wymagań koniecznych do uzyskania oceny dopuszczającej; nie opanował podstawowych wiadomości i umiejętności pozwalających na wykonywanie elementarnych zadań nawet przy istotnym wsparciu nauczyciela.

Uzyskanie oceny celującej nie jest uzależnione od udziału w konkursach lub olimpiadach ani od realizacji treści wykraczających poza program nauczania. Ocenę tę uzyskuje uczeń, który na poziomie realizowanych wymagań wykazuje pełne opanowanie wiadomości i umiejętności oraz wysoki stopień samodzielności, twórczego rozwiązywania problemów, analizy i uzasadniania przyjętych rozwiązań.

Szczegółowe wymagania edukacyjne określono w załącznikach nr 1–7.

2. Formy i obszary sprawdzania osiągnięć edukacyjnych

2.1. Formy sprawdzania

- zadania praktyczne wykonywane przy komputerze;

- praktyczne formy sprawdzania umiejętności;
- sprawdziany pisemne – w rozumieniu Statutu Szkoły, obejmujące co najwyżej trzy ostatnie jednostki lekcyjne;
- kartkówki – obejmujące ostatnią jednostkę lekcyjną;
- testy kompetencji i testy semestralne jako inne formy sprawdzania przewidziane w POU;
- zadania programistyczne i algorytmiczne;
- projekty indywidualne i zespołowe, w tym projekty interdyscyplinarne;
- prezentacje i wyjaśnienia wykonanych rozwiązań;
- odpowiedzi ustne, jeżeli forma ta jest adekwatna do sprawdzanej kompetencji;
- bieżące zadania wykonywane podczas zajęć, jeżeli rzeczywiście pozwalają zweryfikować stopień opanowania określonych wiadomości lub umiejętności.

Kartkówki, sprawdziany i inne formy kontroli obejmujące co najwyżej trzy ostatnie jednostki lekcyjne nie muszą być zapowiadane, zgodnie ze Statutem Szkoły.

2.2. Obszary sprawdzania

- pojęcia i terminologia informatyczna; systemy komputerowe, urządzenia, systemy operacyjne, sieci i bezpieczeństwo;
- tworzenie i redagowanie dokumentów, arkusze kalkulacyjne, grafika 2D i 3D, prezentacje;
- bazy danych, relacyjny model danych i SQL;
- algorytmika i programowanie, w tym przetwarzanie danych liczbowych i tekstowych, iteracja, rekurencja, algorytmy zachłanne, sortowanie i metody numeryczne – odpowiednio do klasy i zakresu;
- technologie internetowe, tworzenie stron i aplikacji, robotyka, gry i aplikacje;
- wizualizacja danych, pliki i dane zewnętrzne, zastosowania interdyscyplinarne;
- projekty, współpraca, dokumentowanie i prezentowanie rezultatów;
- prawo autorskie, bezpieczeństwo cyfrowe, odpowiedzialność i etyka korzystania z technologii.

2.3. Organizacja sprawdzania

Forma sprawdzania powinna odpowiadać charakterowi ocenianej kompetencji. Umiejętności praktyczne powinny być, w miarę możliwości, sprawdzane praktycznie. W programowaniu, bazach danych, grafice, modelowaniu 3D, arkuszu kalkulacyjnym i technologiach WWW nauczyciel może poprosić ucznia o wyjaśnienie sposobu wykonania pracy oraz dokonanie niewielkiej modyfikacji rozwiązania.

2.4. Skala procentowa sprawdzianów pisemnych

Procent uzyskanych punktów	Ocena
0–39%	niedostateczny
powyżej 39% – 52%	dopuszczający
powyżej 52% – 70%	dostateczny
powyżej 70% – 89%	dobry
powyżej 89% i mniej niż 100%	bardzo dobry
100%	celujący

Skala wynika ze Statutu Szkoły. Nie jest automatycznie przenoszona na każdą pracę praktyczną, projektową lub programistyczną; w takich formach mogą być stosowane kryteria jakościowe.

2.5. Minimalna liczba ocen

Tygodniowy wymiar przedmiotu	Minimalna liczba ocen w półroczu
1 godzina	2
2 godziny	3
co najmniej 3 godziny	4

Minimalna liczba ocen wynika ze Statutu Szkoły.

2.6. Samodzielność i korzystanie ze źródeł

Praca ucznia powinna umożliwiać określenie jego rzeczywistych wiadomości i umiejętności. Zakres dopuszczalnego korzystania z Internetu, dokumentacji, materiałów, fragmentów kodu, generatorów treści lub kodu oraz narzędzi sztucznej inteligencji określa nauczyciel odpowiednio do celu zadania. Jeżeli korzystanie z takich narzędzi jest dopuszczone, nauczyciel może wymagać wskazania źródeł oraz wykazania rozumienia, weryfikacji i modyfikacji rozwiązania. W przypadku niesamodzielnej pracy podczas sprawdzianu lub pracy klasowej stosuje się zasady określone w Statucie Szkoły.

3. Kryteria oceniania

3.1. Zasady ogólne

1. Ocena jest dokonywana w odniesieniu do wymagań edukacyjnych właściwych dla danej klasy, zakresu kształcenia i sprawdzanego obszaru.

2. Uwzględnia się przede wszystkim poprawność merytoryczną i techniczną, realizację celu zadania, zastosowanie wiedzy w praktyce, samodzielność, dobór metod i narzędzi, analizę błędów oraz umiejętność wyjaśnienia rozwiązania.
3. Ocena nie jest ustalana wyłącznie na podstawie faktu wykonania zadania; znaczenie mają także sposób rozwiązania, rozumienie i stopień samodzielności.
4. Kryteria stosuje się odpowiednio do charakteru formy sprawdzania; nie wszystkie muszą występować w każdym zadaniu.
5. Stopień trudności i oczekiwany poziom samodzielności są dostosowane do klasy i realizowanego zakresu.

3.2. Zadania praktyczne

Kryterium	Zakres oceny
poprawność	poprawność merytoryczna i techniczna rozwiązania
zgodność z poleceniem	realizacja wymaganych elementów zadania
funkcjonalność	spełnienie zakładanego celu
dobór metod i narzędzi	adekwatność zastosowanych rozwiązań
samodzielność	stopień samodzielnej analizy i wykonania
rozwiązywanie problemów	wykrywanie i usuwanie błędów
rozumienie	wyjaśnienie wykonanych czynności i rozwiązań
jakość opracowania	czytelność, uporządkowanie i przygotowanie rezultatu do zastosowania

Pojedynczy błąd techniczny nie musi prowadzić do istotnego obniżenia oceny, jeżeli nie podważa zasadniczej poprawności rozwiązania, a uczeń potrafi go rozpoznać i usunąć.

3.3. Zadania programistyczne i algorytmiczne

- poprawność analizy problemu i zastosowanego algorytmu;
- prawidłowość implementacji i dobór konstrukcji programistycznych oraz struktur danych;
- działanie programu dla danych testowych i istotnych przypadków szczególnych;
- czytelność i logiczna organizacja rozwiązania;
- umiejętność wykrywania i poprawiania błędów;
- samodzielność, wyjaśnienie działania programu i dokonanie wskazanej modyfikacji.

Samo uzyskanie prawidłowego wyniku programu nie przesądza o ocenie bardzo dobrej lub celującej, jeżeli uczeń nie potrafi wyjaśnić zastosowanego rozwiązania. Jednocześnie pojedynczy błąd składniowy nie musi przesądzać o niskiej ocenie, jeżeli sposób rozwiązania problemu jest prawidłowy, a uczeń potrafi błąd zidentyfikować i poprawić.

3.4. Projekty

- zgodność z założeniami, poprawność merytoryczna i techniczna oraz funkcjonalność;
- sposób planowania, dobór narzędzi i metod oraz stopień samodzielności;
- jakość produktu końcowego i dokumentacji, jeżeli była wymagana;
- prezentacja rezultatu, wyjaśnienie i uzasadnienie zastosowanych rozwiązań.

W projekcie zespołowym nauczyciel może ustalić różne oceny poszczególnym członkom zespołu, jeżeli ich indywidualny wkład lub poziom wykazanych kompetencji były różne.

3.5. Odpowiedzi ustne i wyjaśnienie rozwiązania

Uwzględnia się poprawność merytoryczną, rozumienie pojęć, logiczność wypowiedzi, właściwą terminologię oraz umiejętność zastosowania wiedzy. Błędy językowe lub stylistyczne nie wpływają na ocenę z informatyki, jeżeli nie zmieniają znaczenia wypowiedzi i poprawność językowa nie jest przedmiotem sprawdzanej kompetencji.

3.6. Autorstwo i narzędzia AI

Uczeń przedstawiający pracę jako własną powinien być w stanie wyjaśnić sposób jej wykonania, omówić zastosowane narzędzia, metody lub algorytmy, odpowiedzieć na pytania oraz dokonać adekwatnej modyfikacji. Jeżeli uczeń nie potrafi wyjaśnić przedstawionego rozwiązania, nauczyciel może zweryfikować stopień samodzielnego opanowania umiejętności poprzez dodatkowe pytania lub zadanie praktyczne. Sam fakt wykorzystania narzędzia AI nie stanowi automatycznej podstawy do oceny niedostatecznej; decydują zasady danej formy sprawdzania oraz możliwość potwierdzenia kompetencji ucznia.

3.7. Ocena klasyfikacyjna

Śródroczna i roczna ocena klasyfikacyjna jest wynikiem całościowej oceny stopnia opanowania przez ucznia wiadomości i umiejętności. Nie jest traktowana jako prosta średnia arytmetyczna ocen bieżących. Uwzględnia się w szczególności poziom opanowania kluczowych wymagań, wyniki form obejmujących szerszy zakres materiału, zadania praktyczne, samodzielne rozwiązywanie problemów oraz – odpowiednio do zakresu – kompetencje algorytmiczne, programistyczne i projektowe.

4. Narzędzia i zasady poprawiania ocen

4.1. Zasady ogólne

1. Uczeń ma prawo do poprawiania ocen bieżących na zasadach określonych w Statucie Szkoły i niniejszym POU.
2. Poprawa służy wykazaniu aktualnego, wyższego poziomu opanowania sprawdzanych wiadomości i umiejętności.
3. Poprawa obejmuje te same lub równoważne wymagania co pierwotna forma, ale nie musi wykorzystywać identycznych pytań, danych, plików lub zadań.
4. Termin i sposób poprawy ustala nauczyciel.
5. Forma poprawy powinna odpowiadać charakterowi sprawdzanej kompetencji; umiejętności praktyczne powinny być poprawiane w sposób umożliwiający ich praktyczne wykazanie.

4.2. Prawo do poprawy ocen bieżących

W przypadku przedmiotu realizowanego w wymiarze 1 godziny tygodniowo uczeń ma prawo do poprawienia co najmniej jednej oceny bieżącej w półroczu, a w wymiarze wyższym niż 1 godzina tygodniowo – co najmniej dwóch ocen bieżących w półroczu. Nauczyciel może dopuścić większą liczbę popraw, jeżeli pozwalają na to warunki organizacyjne.

4.3. Termin poprawy

Termin poprawy wyznacza nauczyciel. Poprawy ocen bieżących oraz proponowanej oceny rocznej powinny zakończyć się najpóźniej 10 dni roboczych przed terminem wystawienia rocznej oceny klasyfikacyjnej wskazanym w szkolnym kalendarzu klasyfikacji.

4.4. Poprawa różnych form

- sprawdzian pisemny – obejmuje ten sam zakres wymagań; stosuje się statutową skalę procentową;
- zadanie praktyczne – uczeń wykonuje nowe lub zmodyfikowane zadanie o porównywalnym zakresie albo wskazane elementy pozwalające zweryfikować kompetencję;
- zadanie programistyczne – możliwe jest rozwiązanie nowego problemu, poprawa lub modyfikacja programu, wyjaśnienie algorytmu, testowanie i usuwanie błędów;
- projekt – poprawa może polegać na uzupełnieniu braków, usunięciu błędów, przebudowaniu części rozwiązania, poprawie dokumentacji, ponownej prezentacji lub wykonaniu zadania potwierdzającego określoną umiejętność.

4.5. Nieobecność i zaległe prace

Jeżeli uczeń opuścił pracę pisemną, powinien napisać ją na najbliższej lekcji albo w terminie ustalonym przez nauczyciela. Uzupełnienie pracy, której uczeń nie wykonywał z powodu nieobecności, nie jest poprawą oceny. Zaległe prace pisemne uczeń zalicza w terminie wynikającym ze Statutu Szkoły; termin uzupełnienia innych obowiązkowych form sprawdzania ustala nauczyciel.

4.6. Nieprzygotowanie

Uczeń ma prawo raz w ciągu półrocza zgłosić nieprzygotowanie do lekcji, z wyjątkiem zapowiedzianych form kontroli wiadomości. Nieprzygotowanie należy zgłosić przed rozpoczęciem sprawdzania przygotowania ucznia.

4.7. Wynik poprawy

Wynik poprawy stanowi informację o aktualnym stopniu opanowania sprawdzanych wiadomości i umiejętności. Nie stosuje się zasady mechanicznego obliczania średniej z oceny pierwotnej i oceny uzyskanej w wyniku poprawy. Podczas poprawy obowiązują zasady samodzielności właściwe dla danej formy sprawdzania.

5. Dostosowanie wymagań edukacyjnych

5.1. Zasady ogólne

Nauczyciel dostosowuje wymagania edukacyjne do indywidualnych potrzeb rozwojowych i edukacyjnych oraz możliwości psychofizycznych ucznia w przypadkach określonych przepisami prawa oświatowego i Statutem Szkoły. Zakres i sposób dostosowania ustala się indywidualnie, z uwzględnieniem właściwej opinii, orzeczenia, dokumentacji pomocy psychologiczno-pedagogicznej oraz rozpoznanych potrzeb ucznia.

Dostosowanie może dotyczyć sposobu formułowania poleceń, organizacji stanowiska i pracy, czasu, podziału zadania na etapy, formy odpowiedzi, sposobu prezentowania materiałów, dokumentowania wykonania oraz ograniczenia wpływu trudności niezwiązanych bezpośrednio ze sprawdzaną kompetencją. Dostosowanie nie oznacza automatycznego obniżenia wymagań ani automatycznego podwyższenia oceny.

5.2. Dysleksja

- krótkie, jednoznaczne i uporządkowane polecenia oraz możliwość ich ponownego odczytania lub wyjaśnienia;

- dzielenie rozbudowanych zadań na etapy i – jeżeli jest to uzasadnione – dodatkowy czas;
- ograniczenie znaczenia błędów ortograficznych, literowych i językowych, jeżeli poprawność językowa nie jest celem zadania;
- w programowaniu rozróżnianie błędu wynikającego z niezrozumienia algorytmu od błędu o charakterze literowym lub składniowym, z uwzględnieniem zdolności ucznia do jego identyfikacji i poprawy.

5.3. Dysgrafia

- preferowanie pracy z wykorzystaniem komputera i form elektronicznych;
- nieobniżanie oceny z powodu estetyki pisma odręcznego, jeżeli nie jest przedmiotem oceniania;
- umożliwienie przygotowywania schematów i zapisów za pomocą narzędzi komputerowych.

5.4. Dyskalkulia

- rozdzielanie oceny kompetencji informatycznej od trudności rachunkowych, gdy obliczenia nie są celem sprawdzania;
- możliwość korzystania z danych, wzorów lub informacji pomocniczych, gdy celem jest implementacja, model lub formuła, a nie odtworzenie wzoru;
- dzielenie wieloetapowych problemów obliczeniowych na części oraz możliwość dodatkowej kontroli danych i wyników;
- zachowanie wymagań dotyczących rozumienia algorytmu, formuły, modelu i poprawnej implementacji.

5.5. Spektrum autyzmu

- jednoznaczne i precyzyjne polecenia, ograniczenie niepotrzebnej wieloznaczności;
- przedstawianie struktury zadania, etapów i kryteriów oceny;
- przewidywalna organizacja sprawdzania, a w miarę możliwości ograniczenie nadmiaru bodźców;
- możliwość pracy indywidualnej, jeżeli współpraca nie jest sprawdzaną kompetencją i pozwalają na to warunki organizacyjne;
- odpowiednie dostosowanie czasu lub krótkiej przerwy, jeżeli wynika to z potrzeb ucznia.

5.6. Trudności w koncentracji uwagi, w tym ADHD

- krótkie polecenia, podział zadania na etapy i punkty kontrolne;
- ograniczenie jednoczesnej liczby informacji i możliwość potwierdzenia zrozumienia polecenia;

- odpowiednia organizacja stanowiska, dodatkowy czas lub krótka przerwa – stosownie do potrzeb;
- ocenianie przede wszystkim rezultatu i opanowania kompetencji, a nie tempa organizacji pracy, chyba że czas jest istotnym elementem zadania.

5.7. Ograniczenia wzrokowe

- powiększenie tekstu i elementów interfejsu, zwiększenie czytelności materiałów i wykorzystanie funkcji dostępności;
- materiały w formacie umożliwiającym powiększenie lub odczyt oraz dostosowanie czasu;
- ograniczenie wpływu elementów wyłącznie wizualnych na ocenę, jeżeli nie stanowią istoty sprawdzanej kompetencji.

5.8. Ograniczenia słuchowe

- przekazywanie poleceń również w formie pisemnej i możliwość ponownego zapoznania się z instrukcją;
- prezentowanie najważniejszych informacji w formie tekstowej lub graficznej;
- umożliwienie pisemnego wyjaśnienia rozwiązania zamiast odpowiedzi ustnej, jeżeli forma ustna nie jest celem sprawdzania.

5.9. Ograniczenia sprawności ruchowej

- dostosowanie stanowiska i urządzeń wejściowych oraz wykorzystanie technologii wspomagających w zakresie dostępnych możliwości;
- odpowiednie wydłużenie czasu pracy;
- nietraktowanie szybkości obsługi klawiatury, myszy lub innego urządzenia jako kryterium, jeżeli nie jest przedmiotem sprawdzanej kompetencji.

5.10. Trudności językowe

- krótsze i jednoznaczne polecenia, wyjaśnianie terminologii oraz wspieranie poleceń przykładem lub schematem;
- dodatkowy czas na zapoznanie się z treścią zadania;
- ocenianie przede wszystkim poprawności merytorycznej, jeżeli poprawność językowa nie jest celem zadania.

5.11. Uczeń szczególnie uzdolniony

- zadania o większym stopniu złożoności i możliwość wyboru alternatywnych metod rozwiązania;

- rozszerzenie projektu, optymalizacja rozwiązań i łączenie kilku obszarów informatyki;
- możliwość realizacji autorskich projektów odpowiadających celom kształcenia; zadania rozszerzające nie są warunkiem koniecznym uzyskania oceny celującej.

5.12. Granice i indywidualizacja dostosowania

Wskazane rozwiązania stanowią katalog możliwych sposobów dostosowania i nie są stosowane automatycznie w całości wobec każdego ucznia należącego do danej grupy. Dostosowanie powinno usuwać lub ograniczać barierę utrudniającą wykazanie rzeczywistego poziomu wiadomości i umiejętności. Jeżeli dokumentacja ucznia wymaga modyfikacji zakresu wymagań lub szczególnych rozwiązań, nauczyciel postępuje zgodnie z tymi zaleceniami i obowiązującymi przepisami.

W przypadku zwolnienia ucznia z zajęć informatyki stosuje się odpowiednie postanowienia Statutu Szkoły.

6. Warunki i tryb uzyskiwania rocznej oceny klasyfikacyjnej wyższej niż przewidywana (proponowana)

6.1. Zasady i warunki

Przewidywana roczna ocena klasyfikacyjna nie jest oceną ostateczną. Nauczyciel ustala ją na podstawie stopnia opanowania wiadomości i umiejętności określonych w wymaganiach edukacyjnych. Proponowane oceny klasyfikacyjne są wystawiane w terminie określonym w Statucie Szkoły.

Uczeń, który przystąpił do wszystkich zapowiadanych form kontroli w trakcie roku szkolnego, ma prawo ubiegać się o roczną ocenę klasyfikacyjną wyższą niż przewidywana. Za przystąpienie uznaje się również uzupełnienie obowiązkowej formy opuszczonej z powodu nieobecności w terminie ustalonym przez nauczyciela.

6.2. Zgłoszenie i termin

Uczeń zgłasza nauczycielowi zamiar skorzystania z prawa do uzyskania wyższej oceny po ustaleniu przewidywanej rocznej oceny klasyfikacyjnej, w terminie umożliwiającym przeprowadzenie procedury. Nie wprowadza się dodatkowego, krótszego terminu, jeżeli nie wynika on ze Statutu. Procedura musi zakończyć się najpóźniej 10 dni roboczych przed terminem wystawienia rocznej oceny klasyfikacyjnej określonym w szkolnym kalendarzu.

6.3. Forma i zakres sprawdzianu rocznego

Poprawa przewidywanej oceny rocznej odbywa się poprzez sprawdzian obejmujący materiał z całego roku szkolnego. Ze względu na specyfikę informatyki może on obejmować część pisemną, praktyczną wykonywaną przy komputerze, zadania algorytmiczne lub programistyczne albo połączenie tych form.

Zakres sprawdzianu odpowiada wymaganiom edukacyjnym dla danej klasy, realizowanego zakresu oraz poziomu oceny, o którą ubiega się uczeń. Nie wprowadza się ograniczenia poprawy wyłącznie o jeden stopień, jeżeli takie ograniczenie nie wynika ze Statutu.

6.4. Ocenianie, samodzielność i dostosowania

Do poszczególnych części sprawdzianu stosuje się odpowiednio kryteria określone w punkcie 3. Część pisemna i punktowa podlega statutowej skali procentowej. W części praktycznej ocenia się m.in. poprawność, realizację polecenia, samodzielność, dobór metod, analizę błędów, rozumienie i możliwość modyfikacji rozwiązania.

Sprawdzian jest formą samodzielnego potwierdzenia wiadomości i umiejętności. Uczeń może korzystać wyłącznie z materiałów i narzędzi dopuszczonych przez nauczyciela. Warunki sprawdzianu ucznia objętego dostosowaniem wymagań są odpowiednio dostosowywane zgodnie z punktem 5.

6.5. Wynik

Jeżeli wynik sprawdzianu potwierdza opanowanie wymagań odpowiadających ocenie wyższej niż przewidywana, nauczyciel ustala odpowiednio wyższą roczną ocenę klasyfikacyjną. Nieuzyskanie podczas sprawdzianu wyniku odpowiadającego ocenie wyższej niż przewidywana nie stanowi samoistnej podstawy do obniżenia wcześniej przewidywanej oceny. Roczna ocena klasyfikacyjna jest ostatecznie ustalana przez nauczyciela zgodnie ze Statutem Szkoły i wymaganiami edukacyjnymi.

7. Zasady informowania o ocenach

7.1. Informowanie o zasadach oceniania

Na początku roku szkolnego nauczyciel informuje uczniów i rodziców o wymaganiach edukacyjnych niezbędnych do otrzymania poszczególnych śródrocznych i rocznych ocen klasyfikacyjnych, sposobach sprawdzania osiągnięć oraz warunkach i trybie uzyskania rocznej oceny klasyfikacyjnej wyższej niż przewidywana. Uczniowie są zapoznawani z POU podczas zajęć organizacyjnych.

7.2. Informowanie o ocenach bieżących

Uczeń jest informowany o ocenie po jej ustaleniu. Informacja może być przekazana bezpośrednio podczas zajęć, poprzez wpis do dziennika elektronicznego lub podczas omówienia pracy. Uczeń ma możliwość uzyskania informacji o wiadomościach i umiejętnościach będących przedmiotem sprawdzania oraz elementach pracy, które wpłynęły na ocenę.

7.3. Informowanie rodziców

Rodzice są informowani o postępach w nauce i ocenach za pomocą sposobów przewidzianych w Statucie Szkoły: klasowych spotkań według harmonogramu, indywidualnych spotkań w miarę potrzeb, wpisu oceny do dziennika elektronicznego, wiadomości w dzienniku elektronicznym lub pisma poleconego. Bieżące informacje o ocenach są udostępniane w szczególności za pośrednictwem dziennika elektronicznego, zgodnie z zasadami obowiązującymi w Szkole.

7.4. Oceny przewidywane i proponowane

Na 30 dni kalendarzowych przed rocznym klasyfikacyjnym posiedzeniem Rady Pedagogicznej nauczyciel informuje ucznia i rodziców o przewidywanej rocznej ocenie niedostatecznej poprzez wpis w dzienniku elektronicznym. Na 14 dni roboczych przed śródrocznym oraz rocznym posiedzeniem klasyfikacyjnym Rady Pedagogicznej nauczyciel wystawia proponowane oceny klasyfikacyjne zgodnie ze Statutem Szkoły.

Przewidywana ocena nie jest oceną ostateczną. Uczeń spełniający warunki określone w punkcie 6 może ubiegać się o roczną ocenę wyższą niż przewidywana.

7.5. Jawność i czytelność oceniania

Ocenianie powinno umożliwiać uczniowi rozpoznanie związku: wymaganie edukacyjne → wykonana praca → stopień opanowania wiadomości i umiejętności → uzyskana ocena. W przypadku zadań praktycznych, programistycznych i projektowych nauczyciel może przed rozpoczęciem pracy wskazać szczegółowe kryteria właściwe dla danego zadania.

W sprawach dotyczących informowania o ocenach, które nie zostały szczegółowo uregulowane w niniejszym POU, stosuje się odpowiednie postanowienia Statutu Szkoły oraz obowiązujące przepisy prawa oświatowego.

ZBIORCZE WYMAGANIA EDUKACYJNE NA POSZCZEGÓLNE OCENY

Poniższe zestawienia stanowią syntetyczne ujęcie wymagań edukacyjnych dla poszczególnych klas i poziomów kształcenia. Wymagania przedstawiono łącznie, bez podziału na działy treści. Mają one charakter kumulatywny: uzyskanie wyższej oceny oznacza spełnienie wymagań właściwych dla ocen niższych oraz osiągnięcie wyższego poziomu samodzielności, poprawności, rozumienia problemu, doboru metod i narzędzi, analizy rezultatów oraz uzasadniania własnych rozwiązań.

KLASA I – zakres podstawowy

Ocena dopuszczająca – 2

Uczeń opanował wiadomości i umiejętności konieczne do wykonywania najprostszych zadań informatycznych. Rozpoznaje podstawowe elementy zestawu komputerowego i ich przeznaczenie, zna elementarne zasady bezpiecznej i ergonomicznej pracy, wykonuje podstawowe operacje na plikach i folderach, posługuje się podstawowymi funkcjami oprogramowania, tworzy prosty dokument tekstowy, grafikę rastrową i prezentację. Rozumie pojęcie algorytmu, tworzy lub uzupełnia prosty program w C++ albo Pythonie, wykorzystuje podstawowe dane, wejście i wyjście, proste obliczenia, rozpoznaje instrukcje warunkowe i iteracyjne, uruchamia program i sprawdza jego działanie. Wyszukuje wskazane informacje w Internecie, zna podstawowe zasady bezpieczeństwa cyfrowego i wykonuje prostą część zadania projektowego. Może wymagać niewielkiego wsparcia nauczyciela.

Ocena dostateczna – 3

Uczeń samodzielnie wykonuje typowe zadania o niewielkim stopniu złożoności. Zna podstawowe elementy systemu komputerowego, organizuje pliki i foldery, dobiera podstawowe oprogramowanie, stosuje zasady bezpieczeństwa danych. Tworzy i formatuje dokumenty tekstowe, wykonuje podstawową obróbkę grafiki rastrowej i przygotowuje uporządkowaną prezentację. Analizuje prosty problem i przedstawia rozwiązanie w postaci algorytmu. Samodzielnie tworzy proste programy w C++ i Pythonie, wykorzystując zmienne, typy danych, operatory, wejście i wyjście, instrukcje warunkowe i pętle. Testuje program dla kilku zestawów danych i poprawia proste błędy. Samodzielnie wyszukiuje informacje, stosuje podstawowe zasady ochrony prywatności i realizuje typowy prosty projekt.

Ocena dobra – 4

Uczeń samodzielnie i poprawnie wykonuje typowe zadania informatyczne. Charakteryzuje podstawowe podzespoły komputera i wiąże ich parametry z przeznaczeniem, sprawnie organizuje środowisko pracy. Tworzy poprawnie zorganizowane dokumenty, estetyczne materiały graficzne i logiczne prezentacje, świadomie dobierając poznane narzędzia. Analizuje problem algorytmiczny, dobiera konstrukcje programistyczne, tworzy programy w C++ i Pythonie z instrukcjami warunkowymi i iteracjami, testuje przypadki typowe i wybrane szczególne, rozpoznaje i poprawia typowe błędy. Potrafi wyjaśnić działanie programu i dokonać wskazanej modyfikacji. Ocenia podstawową wiarygodność źródeł, stosuje zasady prawa autorskiego i bezpieczeństwa oraz planuje i dokumentuje prosty projekt.

Ocena bardzo dobra – 5

Uczeń bardzo dobrze opanował wymagania klasy I, sprawnie posługuje się poznanymi technologiami i samodzielnie rozwiązuje problemy o większym stopniu złożoności. Analizuje konfigurację sprzętu, tworzy rozbudowane i poprawne technicznie dokumenty, grafiki i prezentacje, świadomie wykorzystuje zaawansowane spośród poznanych funkcji. Samodzielnie projektuje algorytmy, tworzy poprawne programy w C++ i Pythonie, analizuje przypadki szczególne, przygotowuje dane testowe, diagnozuje błędy logiczne i uzasadnia wybór rozwiązania. Porównuje źródła internetowe, świadomie stosuje zasady bezpieczeństwa i prawa autorskiego oraz realizuje rozbudowany projekt integrujący kilka poznanych umiejętności.

Ocena celująca – 6

Uczeń w pełni opanował wymagania klasy I i wykorzystuje je samodzielnie, świadomie i twórczo. Analizuje wymagania sprzętowe i programowe, porównuje warianty wykonania zadania, tworzy autorskie i dopracowane dokumenty, grafiki oraz prezentacje i krytycznie analizuje rezultat. Samodzielnie formułuje problemy algorytmiczne, projektuje i implementuje rozwiązania, porównuje różne sposoby rozwiązania, analizuje poprawność, przypadki szczególne i podstawowe aspekty efektywności, testuje i optymalizuje rozwiązanie w zakresie poznanych metod. Realizuje autorski projekt integrujący różne umiejętności i szczegółowo uzasadnia decyzje. Ocena celująca nie wymaga konkursów, olimpiad ani treści wykraczających poza program.

Ocena niedostateczna – 1

Uczeń nie osiągnął poziomu koniecznego do uzyskania oceny dopuszczającej. Nawet przy istotnym wsparciu nie potrafi wykonać podstawowych zadań klasy I lub nie rozumie zasadniczego sensu wykonywanych czynności, w szczególności w zakresie organizacji pracy z

komputerem, tworzenia prostych materiałów cyfrowych, elementarnych zasad algorytmiki i programowania oraz bezpiecznego korzystania z technologii.

KLASA I – zakres rozszerzony

Wymagania dla zakresu rozszerzonego mają charakter kumulatywny i obejmują również wymagania przewidziane dla zakresu podstawowego na odpowiednim etapie kształcenia.

Ocena dopuszczająca – 2

Uczeń spełnia wymagania konieczne zakresu podstawowego oraz opanował elementarne umiejętności rozszerzone. Wykorzystuje podstawowe spośród rozszerzonych funkcji edytora tekstu, grafiki i prezentacji, wykonuje proste operacje na sekcjach, szablonach, warstwach, filtrach i animacjach. Rozpoznaje konstrukcje programowania graficznego i modyfikuje prosty projekt w Scratchu, rozpoznaje elementy aplikacji okienkowej w Lazarusie, tworzy proste programy obliczeniowe w C++ i Pythonie oraz wykonuje proste zadanie projektowe.

Ocena dostateczna – 3

Uczeń spełnia wymagania na ocenę dostateczną w zakresie podstawowym i samodzielnie wykonuje typowe zadania rozszerzone. Wykorzystuje rozszerzone możliwości edytora, grafiki i prezentacji, tworzy prosty projekt w Scratchu ze zmiennymi i iteracją, prostą aplikację okienkową w Lazarusie z komponentami i obsługą zdarzeń oraz implementuje typowe problemy matematyczne i obliczeniowe w C++ i Pythonie. Testuje programy, usuwa proste błędy i realizuje typowe zadanie projektowe.

Ocena dobra – 4

Uczeń spełnia wymagania na ocenę dobrą w zakresie podstawowym i samodzielnie wykorzystuje rozszerzone narzędzia do problemów typowych i częściowo nowych. Świadomie dobiera zaawansowane funkcje programów, projektuje algorytmy graficzne i proste modele w Scratchu, samodzielnie projektuje aplikację okienkową w Lazarusie, analizuje problemy matematyczne lub użytkowe i implementuje je w C++ lub Pythonie, stosuje zagnieżdżone konstrukcje sterujące, porównuje podstawowe sposoby realizacji w obu językach, testuje i wyjaśnia program oraz planuje etapy projektu.

Ocena bardzo dobra – 5

Uczeń spełnia wymagania na ocenę bardzo dobrą w zakresie podstawowym i samodzielnie rozwiązuje bardziej złożone problemy rozszerzone. Tworzy rozbudowane dokumenty, grafiki i prezentacje, projektuje złożone algorytmy graficzne i modele matematyczne lub fizyczne, tworzy funkcjonalne aplikacje okienkowe, samodzielnie projektuje rozwiązania w C++ i Pythonie, analizuje przypadki szczególne, porównuje implementacje, diagnozuje błędy i realizuje rozbudowany projekt integrujący kilka technologii.

Ocena celująca – 6

Uczeń spełnia wymagania na ocenę celującą w zakresie podstawowym i w pełni twórczo wykorzystuje wymagania rozszerzone. Projektuje profesjonalnie zorganizowane materiały, autorskie projekty graficzne i algorytmy, tworzy własne aplikacje w Lazarusie, samodzielnie formułuje problemy obliczeniowe i projektuje rozwiązania w C++ i Pythonie, porównuje algorytmy i implementacje, analizuje poprawność i podstawową efektywność, tworzy zestawy testów, optymalizuje kod i przenosi idee między językami. Realizuje autorski projekt integrujący różne technologie. Ocena 6 nie wymaga treści ponadprogramowych ani osiągnięć konkursowych.

Ocena niedostateczna – 1

Uczeń nie spełnia wymagań koniecznych zakresu podstawowego i rozszerzonego. Nawet przy istotnym wsparciu nie potrafi zastosować podstawowych funkcji poznanego oprogramowania, elementarnych konstrukcji programistycznych, uruchomić i sprawdzić prostego rozwiązania albo wyjaśnić zasadniczego sensu przedstawionej pracy.

KLASA II – zakres podstawowy

Ocena dopuszczająca – 2

Uczeń opanował wymagania konieczne klasy II. Zna podstawowe zasady bezpieczeństwa sieci i ochrony danych, wykonuje proste obliczenia i podstawowe operacje na danych w arkuszu, tworzy prostą grafikę wektorową, stosuje podstawowe pętle w C++ i Pythonie, rozpoznaje funkcje, tworzy prostą tablicę lub listę, rozumie ideę badania pierwszości, ciągu Fibonacciego i prostych sortowań. Zna podstawy prawa autorskiego i netykiety, korzysta z poczty i chmury, tworzy prostą stronę HTML/CSS według wzoru i wykonuje prostą część projektu.

Ocena dostateczna – 3

Uczeń samodzielnie wykonuje typowe zadania klasy II. Stosuje zasady bezpieczeństwa, tworzy formuły i podstawowe funkcje arkusza, sortuje, filtruje i organizuje dane, tworzy prostą grafikę wektorową. Stosuje pętle, tworzy proste funkcje, pracuje z tablicami i listami, implementuje proste algorytmy badania pierwszości, Fibonacciego i sortowania, testuje programy. Rozumie zasady prawa autorskiego, poprawnie korzysta z komunikacji cyfrowej, tworzy prostą stronę HTML/CSS z podstawowym elementem JavaScript i realizuje typowy projekt.

Ocena dobra – 4

Uczeń samodzielnie i poprawnie wykonuje typowe zadania klasy II. Analizuje zagrożenia i dobiera środki ochrony, projektuje uporządkowany arkusz, stosuje funkcje, filtrowanie i tabele przestawne oraz interpretuje wyniki. Tworzy złożone obiekty wektorowe, dobiera rodzaj pętli, dzieli program na funkcje, przetwarza tablice i listy, analizuje algorytmy liczbowe i sortowania oraz usuwa błędy logiczne. Analizuje sytuacje prawne, świadomie organizuje komunikację cyfrową, tworzy wieloelementową stronę z HTML, CSS i JavaScript oraz planuje etapy projektu.

Ocena bardzo dobra – 5

Uczeń bardzo dobrze opanował wymagania klasy II. Analizuje warianty zabezpieczeń, tworzy rozbudowane arkusze analityczne, łączy funkcje, filtrowanie i tabele przestawne, kontroluje obliczenia i wyciąga wnioski. Projektuje rozbudowaną grafikę wektorową, tworzy złożone algorytmy iteracyjne i modularne programy, rozwiązuje zadania z tablicami i listami, analizuje efektywność prostych algorytmów i porównuje sortowania. Samodzielnie diagnozuje błędy, analizuje problemy prawne, projektuje spójny serwis WWW i realizuje rozbudowany projekt.

Ocena celująca – 6

Uczeń w pełni opanował wymagania klasy II i wykorzystuje je twórczo. Projektuje strategie bezpieczeństwa, autorskie modele arkuszowe, analizuje warianty organizacji danych i optymalizuje obliczenia. Tworzy autorską grafikę wektorową, porównuje algorytmy iteracyjne, projektuje strukturę funkcji i kolekcji, analizuje podstawowy koszt obliczeniowy, proponuje ulepszenia algorytmów, przenosi rozwiązania między C++ i Pythonem, tworzy testy i optymalizuje kod. Przygotowuje autorskie analizy problemów prawnych lub bezpieczeństwa, projektuje serwis internetowy i projekt integrujący kilka obszarów. Ocena 6 nie wymaga treści ponadprogramowych ani konkursów.

Ocena niedostateczna – 1

Uczeń nie osiągnął poziomu koniecznego do oceny dopuszczającej. Nawet przy istotnym wsparciu nie potrafi wykonać podstawowych zadań z bezpieczeństwa sieci, arkusza kalkulacyjnego, grafiki wektorowej, iteracji, funkcji, tablic lub list, podstawowych algorytmów oraz tworzenia elementarnej strony internetowej.

KLASA II – zakres rozszerzony

Wymagania dla zakresu rozszerzonego mają charakter kumulatywny i obejmują również wymagania przewidziane dla zakresu podstawowego na odpowiednim etapie kształcenia.

Ocena dopuszczająca – 2

Uczeń spełnia wymagania konieczne zakresu podstawowego oraz elementarne wymagania rozszerzone. Rozpoznaje problemy bezpieczeństwa, stosuje podstawowe funkcje statystyczne, warunkowe, tekstowe lub daty/czas według wzoru, importuje i porządkuje dane, tworzy prosty wykres, filtruje i korzysta z tabel przestawnych według instrukcji. Wykonuje podstawowe operacje grafiki wektorowej, implementuje proste iteracje, funkcje i operacje na tablicach/listach, rozpoznaje reprezentację binarną, ideę numerycznego wyznaczania pierwiastków i realizuje prosty projekt.

Ocena dostateczna – 3

Uczeń spełnia wymagania dostateczne zakresu podstawowego i samodzielnie wykonuje typowe zadania rozszerzone. Stosuje funkcje arkusza, przygotowuje dane, tworzy wykresy kilku serii, proste linie trendu i tabele przestawne. Tworzy bardziej rozbudowaną grafikę, implementuje algorytmy w C++ i Pythonie, wykorzystuje funkcje i kolekcje, wykonuje konwersje reprezentacji liczb, implementuje metodę bisekcji, rozumie kryterium zakończenia, tworzy programy matematyczne, wykresy funkcji, podstawową interpolację i prosty projekt WWW.

Ocena dobra – 4

Uczeń spełnia wymagania dobre zakresu podstawowego i samodzielnie stosuje metody rozszerzone w problemach typowych i częściowo nowych. Łączy funkcje arkusza, przetwarza dane importowane, tworzy wykresy wieloseryjne, stosuje linię trendu i interpretuje R^2 , tworzy złożone tabele przestawne i modele. Świadomie porównuje konstrukcje w C++ i Pythonie, tworzy funkcje, przetwarza tablice/listy, analizuje reprezentację liczb, implementuje kilka metod wyznaczania pierwiastków, porównuje dokładność i liczbę iteracji, realizuje problemy obliczeniowe, interpolację, wizualizację i projekt internetowy.

Ocena bardzo dobra – 5

Uczeń spełnia wymagania bardzo dobre zakresu podstawowego i samodzielnie rozwiązuje złożone problemy rozszerzone. Projektuje rozbudowane arkusze analityczne, przetwarza dane zewnętrzne, dobiera wykres i model aproksymacyjny, interpretuje współczynnik determinacji, tworzy modele interdyscyplinarne i analizuje wpływ parametrów. Projektuje modularne programy, świadomie wykorzystuje funkcje i struktury danych, analizuje zakres i

dokładność reprezentacji liczb, implementuje i porównuje metody numeryczne, ich zbieżność i sytuacje problemowe, realizuje złożone projekty matematyczno-programistyczne, wizualizacyjne i WWW.

Ocena celująca – 6

Uczeń spełnia wymagania celujące zakresu podstawowego i twórczo wykorzystuje zakres rozszerzony. Projektuje strategie bezpieczeństwa i rozbudowane modele danych, identyfikuje anomalie, optymalizuje obliczenia, porównuje modele aproksymacyjne i krytycznie interpretuje ich jakość. Tworzy modele interdyscyplinarne, analizuje efektywność algorytmów, konsekwencje reprezentacji liczb i projektuje programy porównujące metody numeryczne. Bada wpływ parametrów, tolerancji i kryteriów zatrzymania, analizuje ograniczenia metod, tworzy autorskie projekty matematyczno-programistyczne, wizualizacyjne i WWW oraz uzasadnia i optymalizuje rozwiązania. Ocena 6 nie wymaga materiału ponadprogramowego ani konkursów.

Ocena niedostateczna – 1

Uczeń nie spełnia wymagań koniecznych zakresu podstawowego i rozszerzonego. Nie potrafi poprawnie wykorzystać podstawowych funkcji arkusza, wykonać prostego przetwarzania danych, zastosować elementarnych konstrukcji programistycznych, pracować na prostych strukturach danych, wyjaśnić idei poznanego algorytmu lub metody numerycznej ani uruchomić i zweryfikować prostego rozwiązania.

KLASA III – zakres podstawowy

Ocena dopuszczająca – 2

Uczeń opanował wymagania konieczne klasy III. Rozpoznaje system dwójkowy i szesnastkowy i wykonuje proste konwersje według wzoru, wskazuje zastosowania urządzeń cyfrowych, wykonuje proste zadania grafiki 2D i tworzy prosty model 3D. Zna podstawowe pojęcia baz danych, wprowadza dane i wykonuje proste operacje. Śledzi algorytmy konwersji, wykonuje podstawowe operacje na tekstach, rozpoznaje ideę szyfrowania, rekurencji i algorytmu zachłannego, wykonuje proste zadanie robotyczne, modyfikuje prostą stronę internetową i wykonuje część projektu.

Ocena dostateczna – 3

Uczeń samodzielnie wykonuje typowe zadania klasy III. Wykonuje konwersje systemów liczbowych, opisuje urządzenia cyfrowe, tworzy proste projekty 2D i modele 3D. Tworzy prostą tabelę bazy, rozpoznaje klucz, tworzy formularz, kwerendę lub raport, sortuje i filtruje dane. Implementuje proste konwersje, algorytmy tekstowe i szyfrujące, prostą funkcję rekurencyjną i algorytm zachłanny. Realizuje prosty scenariusz robotyczny, tworzy stronę HTML/CSS i wykorzystuje podstawowe elementy poznanych technologii internetowych oraz realizuje typowy projekt.

Ocena dobra – 4

Uczeń samodzielnie i poprawnie wykonuje typowe zadania klasy III. Sprawnie wykonuje konwersje, porównuje zastosowania urządzeń, dobiera technikę grafiki 2D i tworzy modele 3D z kilku elementów. Projektuje prostą relacyjną bazę kilku tabel, relacje, kwerendy, formularze i raporty. Implementuje konwersje, algorytmy tekstowe i szyfrujące, wyjaśnia rekurencję i projektuje prosty algorytm rekurencyjny, rozwiązuje typowy problem zachłanny. Modyfikuje sterowanie układem robotycznym, tworzy serwis internetowy z kilkoma technologiami, testuje i usuwa błędy oraz planuje projekt.

Ocena bardzo dobra – 5

Uczeń bardzo dobrze opanował wymagania klasy III. Analizuje systemy liczbowe i zastosowania urządzeń, tworzy rozbudowaną grafikę 2D i bardziej złożone modele 3D. Analizuje wymagania bazy danych, projektuje jej strukturę, dobiera klucze i relacje oraz ogranicza redundancję. Projektuje algorytmy konwersji, przetwarzania tekstów i szyfrowania, porównuje rekurencję z iteracją, analizuje zasadność strategii zachłannej, projektuje i testuje projekt robotyczny, tworzy spójny serwis internetowy i realizuje rozbudowany projekt integrujący kilka obszarów.

Ocena celująca – 6

Uczeń w pełni opanował wymagania klasy III i wykorzystuje je twórczo. Tworzy własne rozwiązania konwersji, krytycznie analizuje technologie cyfrowe, projektuje autorską grafikę 2D/3D i relacyjną bazę danych, analizuje alternatywne struktury. Tworzy uniwersalne algorytmy konwersji, tekstowe i szyfrujące, autorskie rozwiązania rekurencyjne i zachłanne oraz analizuje ich ograniczenia. Projektuje własny algorytm sterowania robotem, autorski projekt internetowy i projekt integrujący kilka obszarów, porównuje rozwiązania i optymalizuje rezultat. Ocena 6 nie wymaga konkursów ani treści ponadprogramowych.

Ocena niedostateczna – 1

Uczeń nie osiągnął poziomu koniecznego do oceny dopuszczającej. Nie potrafi nawet przy istotnym wsparciu wykonać podstawowych zadań dotyczących systemów liczbowych, grafiki 2D/3D, baz danych, tekstów, rekurencji, strategii zachłannych, robotyki lub podstawowego projektu internetowego.

KLASA III – zakres rozszerzony

Wymagania dla zakresu rozszerzonego mają charakter kumulatywny i obejmują również wymagania przewidziane dla zakresu podstawowego na odpowiednim etapie kształcenia.

Ocena dopuszczająca – 2

Uczeń spełnia wymagania konieczne zakresu podstawowego oraz elementarne wymagania rozszerzone. Wykonuje podstawowe operacje w bardziej zaawansowanej grafice 2D/3D, rozpoznaje strukturę relacyjnej bazy danych, uruchamia gotowe zapytanie SQL i rozpoznaje jego składnię. Rozróżnia podstawowe reprezentacje liczb, rozumie ideę numerycznego wyznaczania pierwiastka, wykonuje prosty szyfr, rozpoznaje przypadek bazowy rekurencji i śledzi prosty algorytm zachłanny. Wykonuje podstawowe zadanie robotyczne, prosty fragment projektu internetowego i projektu integrującego.

Ocena dostateczna – 3

Uczeń spełnia wymagania dostateczne zakresu podstawowego i samodzielnie wykonuje typowe zadania rozszerzone. Realizuje projekty 2D i modele 3D, tworzy tabele, klucze i relacje, proste zapytania SQL z warunkiem i sortowaniem. Wykonuje konwersje liczb i analizuje reprezentację zmiennoprzecinkową, implementuje podstawową metodę numeryczną, prosty szyfr i operacje tekstowe, funkcję rekurencyjną i algorytm zachłanny. Przygotowuje sekwencję działania układu robotycznego, tworzy prosty serwis internetowy i realizuje typowy projekt.

Ocena dobra – 4

Uczeń spełnia wymagania dobre zakresu podstawowego i samodzielnie stosuje metody rozszerzone do problemów typowych i częściowo nowych. Łączy techniki grafiki 2D, projektuje modele 3D, strukturę kilku tabel i zapytania SQL z kilkoma warunkami i danymi z kilku tabel. Analizuje zapis liczb, implementuje algorytmy reprezentacji, kilka metod numerycznych i porównuje ich działanie. Tworzy programy szyfrujące, wyjaśnia różnicę szyfrowania symetrycznego i asymetrycznego, tworzy rozwiązania rekurencyjne i zachłanne, steruje układem robotycznym oraz planuje i testuje serwis internetowy.

Ocena bardzo dobra – 5

Uczeń spełnia wymagania bardzo dobre zakresu podstawowego i samodzielnie rozwiązuje złożone problemy rozszerzone. Projektuje rozbudowaną grafikę i modele 3D, kompletną relacyjną bazę z ograniczeniem redundancji oraz zestawu zapytań SQL. Analizuje zakres i dokładność reprezentacji liczb, konsekwencje arytmetyki zmiennoprzecinkowej, implementuje i porównuje metody numeryczne, ich zbieżność i dokładność. Projektuje

przetwarzanie i szyfrowanie tekstu, analizuje metody kryptograficzne, wykorzystuje rekurencję w problemach graficznych lub numerycznych, projektuje rozwiązania zachłanne, projekt robotyczny, rozbudowany serwis WWW i projekt integrujący.

Ocena celująca – 6

Uczeń spełnia wymagania celujące zakresu podstawowego i twórczo wykorzystuje wymagania rozszerzone. Projektuje autorskie projekty 2D/3D, relacyjne bazy danych i rozbudowane zapytania SQL, analizuje alternatywne sposoby uzyskania danych. Bada wpływ reprezentacji liczb na wyniki, projektuje eksperymenty numeryczne, programy porównujące metody, analizuje tolerancję, zbieżność i ograniczenia. Tworzy autorskie projekty kryptograficzne, rekurencyjne i zachłanne, projektuje sterowanie robotem, autorski projekt internetowy i projekt integrujący, porównuje rozwiązania i optymalizuje je. Ocena 6 nie wymaga materiału ponadprogramowego ani konkursów.

Ocena niedostateczna – 1

Uczeń nie spełnia wymagań koniecznych zakresu podstawowego i rozszerzonego. Nie potrafi wykonać podstawowych operacji w relacyjnej bazie, odczytać lub utworzyć prostego SQL, wykonać elementarnej konwersji, wyjaśnić idei metody numerycznej, zastosować prostego algorytmu tekstowego, rekurencyjnego lub zachłannego, uruchomić podstawowego projektu robotycznego ani przygotować prostego rozwiązania internetowego.

KLASA IV – zakres rozszerzony

Wymagania dla zakresu rozszerzonego mają charakter kumulatywny i obejmują również wymagania przewidziane dla zakresu podstawowego na odpowiednim etapie kształcenia.

Ocena dopuszczająca – 2

Uczeń opanował wiadomości i umiejętności konieczne klasy IV. Wykonuje prostą grafikę rastrową lub wektorową, prosty model 3D, wielostronicowy dokument i proste obliczenia w arkuszu lub oprogramowaniu specjalistycznym. Przygotowuje prostą prezentację produktu, uruchamia i modyfikuje prosty projekt gry, rozpoznaje elementy aplikacji obiektowej, odczytuje prosty zestaw danych, wykonuje podstawowy odczyt lub zapis pliku, modyfikuje prostą stronę internetową i wykonuje wyznaczoną część projektu. Może wymagać niewielkiego wsparcia.

Ocena dostateczna – 3

Uczeń samodzielnie wykonuje typowe zadania klasy IV. Tworzy projekt grafiki 2D i model 3D, uporządkowany dokument wielostronicowy ze stylami, nagłówkami, numeracją i spisem treści. Tworzy arkusz dla typowego problemu matematycznego lub fizycznego, prezentację produktu i opracowanie obliczeniowe w poznanym oprogramowaniu. Tworzy prosty mechanizm gry, aplikację okienkową w Lazarusie, wczytuje i wizualizuje dane, odczytuje i zapisuje pliki, tworzy prosty serwis internetowy oraz realizuje typowy projekt i wyjaśnia jego działanie.

Ocena dobra – 4

Uczeń samodzielnie i poprawnie wykonuje typowe zadania klasy IV i dostosowuje rozwiązania do zmienionych warunków. Świadomie dobiera technikę grafiki 2D, projektuje modele 3D, przygotowuje dokument o strukturze technicznej lub naukowej z automatycznymi elementami, projektuje arkusz interdyscyplinarny i wizualizację wyników. Tworzy prezentację produktu, uporządkowane opracowanie obliczeniowe, logikę gry i aplikację okienkową z kontrolą danych. Projektuje wizualizację, przetwarza wiele rekordów z pliku, sprawdza poprawność danych, tworzy bardziej złożony serwis WWW, testuje i usuwa błędy oraz planuje projekt.

Ocena bardzo dobra – 5

Uczeń bardzo dobrze opanował wymagania klasy IV. Projektuje rozbudowaną grafikę 2D i zaawansowane modele 3D, tworzy dokumenty techniczne lub zbliżone do pracy dyplomowej i kontroluje ich strukturę. Tworzy rozbudowane modele arkuszowe, analizuje wpływ danych i ograniczenia modelu, projektuje prezentację reklamową i złożone opracowania

matematyczne/fizyczne. Realizuje kompletną prostą grę, funkcjonalne aplikacje obiektowe, narzędzia wizualizacji i aplikacje pracujące z danymi zewnętrznymi. Analizuje sytuacje wyjątkowe, tworzy funkcjonalny serwis internetowy oraz rozbudowany projekt integrujący kilka obszarów, dokumentuje go i uzasadnia decyzje.

Ocena celująca – 6

Uczeń w pełni opanował wymagania klasy IV i wykorzystuje je samodzielnie, twórczo i integrująco. Tworzy autorskie projekty 2D i 3D, porównuje strategie modelowania i optymalizuje modele. Projektuje profesjonalne dokumenty i szablony, autorskie modele arkuszowe oraz matematyczne/fizyczne, analizuje warianty i weryfikuje wyniki. Projektuje autorską prezentację produktu i grę, aplikacje obiektowe, narzędzia wizualizacji i rozwiązania przetwarzające dane zewnętrzne. Tworzy autorskie projekty internetowe, analizuje funkcjonalność, ergonomię i bezpieczeństwo oraz realizuje projekt końcowy integrujący kilka obszarów. Porównuje technologie, testuje, krytycznie ocenia i optymalizuje rozwiązania. Ocena 6 nie wymaga konkursów, olimpiad ani treści wykraczających poza program.

Ocena niedostateczna – 1

Uczeń nie osiągnął poziomu koniecznego do oceny dopuszczającej. Nawet przy istotnym wsparciu nie potrafi wykonać podstawowych zadań dotyczących grafiki 2D/3D, organizacji dokumentów, prostych obliczeń, elementarnych aplikacji i projektów programistycznych, odczytu i zapisu danych zewnętrznych, podstawowego rozwiązania internetowego lub własnej części projektu.

POSTANOWIENIA KOŃCOWE

Dokument stosuje się w roku szkolnym 2026/2027 w zakresie zgodnym ze Statutem Szkoły, obowiązującymi przepisami prawa oraz realizowanymi programami i planami nauczania informatyki. W sprawach nieuregulowanych niniejszym dokumentem stosuje się aktualne postanowienia Statutu Szkoły i obowiązujące przepisy prawa oświatowego.

Zbiorcze wymagania edukacyjne stanowią praktyczne i syntetyczne rozwinięcie zasad oceniania określonych w części głównej POU i służą w szczególności do przekazywania uczniom i rodzicom czytelnej informacji o poziomie osiągnięć wymaganym na poszczególne oceny.

PROPOZYCJA TEMATYKI ZAJĘĆ Z INFORMATYKI

Założenia dokumentu

Dokument przedstawia propozycję organizacji treści kształcenia z informatyki w klasach I–IV liceum ogólnokształcącego. Ma charakter pomocniczy i elastyczny. Nauczyciel może zmieniać kolejność realizacji bloków i tematów, łączyć lub dzielić poszczególne zagadnienia, modyfikować liczbę przeznaczonych na nie godzin oraz dobierać oprogramowanie i środowiska informatyczne odpowiednio do wyposażenia szkoły, potrzeb uczniów i przyjętych metod pracy, z zachowaniem obowiązujących wymagań programowych.

Tematy sformułowano przede wszystkim w języku umiejętności i problemów informatycznych, a nie nazw konkretnych programów. Nazwy środowisk podano jedynie jako przykłady. Pozwala to wykorzystać materiał niezależnie od systemu operacyjnego, wyposażenia pracowni i aktualnie stosowanego oprogramowania.

Podane liczby godzin mają charakter orientacyjny. W klasach I–III przyjęto model 36 godzin dla zakresu podstawowego i 72 godzin dla zakresu rozszerzonego. W klasie IV przedstawiono propozycję 72 godzin dla zakresu rozszerzonego. Część godzin pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje i dostosowanie tempa pracy do zespołu klasowego.

KLASA I – ZAKRES PODSTAWOWY

Orientacyjny wymiar: 36 godzin. W poniższej propozycji przypisano 31 godzin do tematów i bloków, a 5 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Organizacja pracy i bezpieczeństwo

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	Organizacja zajęć, zasady pracy w pracowni, BHP i zasady oceniania	1	organizacja pracy, bezpieczeństwo
2	Diagnoza umiejętności informatycznych uczniów	1	zadania diagnostyczne

System komputerowy

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
3	Budowa komputera i podstawowe parametry podzespołów	1	analiza konfiguracji
4	Urządzenia wejścia i wyjścia oraz dobór urządzeń do zastosowania	1	zadania praktyczne
5	Zakup, modernizacja i prawidłowa eksploatacja komputera	1	analiza ofert i potrzeb użytkownika

Dokumenty i multimedia

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
6	Tworzenie i formatowanie dokumentów tekstowych	2	style, akapity, listy, tabele
7	Grafika w dokumencie i przygotowanie dokumentu wielostronicowego	2	nagłówki, stopki, numeracja
8	Podstawy grafiki rastrowej	2	np. GIMP lub inne narzędzie
9	Warstwy, zaznaczenia, korekcja i przekształcenia obrazu	2	zadania praktyczne
10	Projektowanie prezentacji multimedialnej	2	struktura, tekst, grafika
11	Animacje, przejścia i zasady czytelnej prezentacji	1	projekt prezentacyjny

Algorytmika i programowanie

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
12	Od problemu do algorytmu – sposoby opisu rozwiązania	1	analiza problemu
13	Pierwszy program – dane, zmienne, wejście i wyjście	2	C++ lub Python
14	Operatory i proste obliczenia	2	zadania obliczeniowe
15	Instrukcja warunkowa	2	problemy decyzyjne
16	Iteracja – powtarzanie czynności w algorytmie	3	pętle i testowanie
17	Testowanie programu i poprawianie błędów	1	debugowanie

Internet i projekty

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
18	Skuteczne wyszukiwanie i ocena informacji w Internecie	1	wiarygodność źródeł
19	Usługi internetowe, prywatność i bezpieczeństwo kont	1	praktyka cyfrowa
20	Mini-projekt integrujący poznane narzędzia	2	praca indywidualna lub zespołowa

KLASA I – ZAKRES ROZSZERZONY

Orientacyjny wymiar: 72 godzin. W poniższej propozycji przypisano 63 godzin do tematów i bloków, a 9 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Systemy i oprogramowanie

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	System operacyjny i organizacja środowiska pracy	2	konfiguracja i narzędzia systemowe
2	Dobór oprogramowania i formatów danych do zadania	1	analiza zastosowań

Zaawansowane dokumenty i grafika

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
3	Zaawansowana struktura dokumentu tekstowego	3	sekcje, style, szablony
4	Automatyzacja i organizacja rozbudowanego dokumentu	3	elementy automatyczne
5	Zaawansowana grafika rastrowa	4	warstwy, selekcje, filtry, transformacje
6	Projekt graficzny – od koncepcji do publikacji	3	zadanie autorskie
7	Zaawansowana prezentacja multimedialna	3	animacja, multimedia, spójność przekazu

Programowanie graficzne i modelowanie

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
8	Algorytmy w środowisku programowania graficznego	3	np. Scratch
9	Fraktale i grafika generowana algorytmicznie	3	iteracja i geometria
10	Proste modele matematyczne i fizyczne w programowaniu graficznym	3	symulacje

Aplikacje okienkowe

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
11	Wprowadzenie do aplikacji okienkowych	2	np. Lazarus
12	Komponenty, właściwości i zdarzenia	3	interfejs użytkownika
13	Podstawy programowania obiektowego w praktyce	3	proste aplikacje

C++ i Python – rozwiązywanie problemów

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
14	Porównanie podstawowych konstrukcji C++ i Pythona	2	składnia i sposób zapisu
15	Problemy obliczeniowe i matematyczne	4	projektowanie algorytmów
16	Instrukcje warunkowe – zadania złożone	3	przypadki i walidacja
17	Pętle i zagnieżdżone iteracje	5	analiza przebiegu
18	Testowanie, diagnozowanie i modyfikowanie programów	3	debugowanie
19	Porównanie różnych sposobów rozwiązania problemu	2	poprawność i podstawowa efektywność

Internet i projekty

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
20	Praca zespołowa z wykorzystaniem narzędzi internetowych	2	organizacja współpracy
21	Projekt integrujący programowanie i multimedia	4	projekt semestralny
22	Dokumentowanie i prezentacja rozwiązania	2	opis, testy, wnioski

KLASA II – ZAKRES PODSTAWOWY

Orientacyjny wymiar: 36 godzin. W poniższej propozycji przypisano 34 godzin do tematów i bloków, a 2 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Sieci i bezpieczeństwo

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	Bezpieczeństwo danych i użytkownika w sieci	2	zagrożenia i zabezpieczenia
2	Ochrona kont, danych i kopie zapasowe	1	praktyczne scenariusze

Arkusz kalkulacyjny i analiza danych

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
3	Formuły, adresowanie i podstawowe funkcje arkusza	2	zadania praktyczne
4	Sortowanie, filtrowanie i organizacja danych	1	praca z tabelą
5	Wykresy i podstawowa interpretacja danych	1	wizualizacja
6	Tabele przestawne jako narzędzie podsumowania danych	1	analiza zestawu danych

Grafika wektorowa

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
7	Podstawy grafiki wektorowej	2	np. Inkscape
8	Ścieżki, warstwy i przekształcenia obiektów	2	projekt graficzny

Programowanie i algorytmy

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
9	Iteracja w C++ i Pythonie – utrwalenie i zastosowania	2	dobór pętli
10	Funkcje – podział problemu na mniejsze części	2	parametry i wyniki
11	Tablice w C++ i listy w Pythonie	3	przetwarzanie kolekcji
12	Badanie pierwszości liczby	1	algorytm liczbowy
13	Ciąg Fibonacciego i problemy iteracyjne	1	implementacja
14	Podstawowe algorytmy sortowania	3	implementacja i porównanie
15	Testowanie i diagnozowanie błędów	1	dane testowe

Prawo i odpowiedzialność cyfrowa

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
16	Prawo autorskie, licencje i legalne korzystanie z zasobów	1	studia przypadków
17	Ochrona wizerunku, odpowiedzialność i przestępczość komputerowa	1	dyskusja problemowa
18	Poczta elektroniczna, netykieta i usługi chmurowe	1	organizacja komunikacji

Technologie internetowe

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
19	Struktura dokumentu HTML	1	semantyczna budowa strony
20	CSS – formatowanie i układ strony	2	oddzielenie treści od prezentacji
21	Wprowadzenie do interaktywności strony	1	podstawy JavaScript

22	Projekt prostej witryny internetowej	2	integracja HTML/CSS/JS
----	--------------------------------------	---	------------------------

KLASA II – ZAKRES ROZSZERZONY

Orientacyjny wymiar: 72 godzin. W poniższej propozycji przypisano 70 godzin do tematów i bloków, a 2 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Sieci i bezpieczeństwo

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	Analiza zagrożeń sieciowych i strategię ochrony danych	3	scenariusze i ocena ryzyka
2	Bezpieczna organizacja pracy w środowisku sieciowym	2	uprawnienia, kopie, dobre praktyki

Zaawansowany arkusz i analiza danych

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
3	Funkcje warunkowe, statystyczne, tekstowe oraz daty i czasu	4	łączenie funkcji
4	Importowanie, czyszczenie i przygotowanie danych	3	dane zewnętrzne
5	Wykresy wieloseryjne i dodatkowe osie	2	dobór wizualizacji
6	Linie trendu, regresja i interpretacja współczynnika R^2	4	modelowanie danych
7	Tabele przestawne i analiza wielowymiarowa	3	podsumowania
8	Model interdyscyplinarny w arkuszu	4	np. matematyka lub fizyka

Grafika wektorowa

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
9	Zaawansowane operacje na ścieżkach i obiektach	3	projekt użytkowy
10	Autorski projekt grafiki wektorowej	3	warianty i ocena rozwiązania

Programowanie i struktury danych

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
11	Złożone konstrukcje iteracyjne	3	C++ i Python
12	Funkcje i modularna organizacja programu	3	dekompozycja
13	Tablice i listy – przetwarzanie danych	4	wyszukiwanie i agregacja
14	Reprezentacja liczb w komputerze	3	systemy pozycyjne i ograniczenia
15	Dokładność reprezentacji zmiennoprzecinkowej	2	eksperymenty obliczeniowe

Metody numeryczne i wizualizacja

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
16	Numeryczne wyznaczanie pierwiastków – metoda bisekcji	3	implementacja
17	Porównanie metod rozwiązywania równań	4	dokładność i liczba iteracji
18	Kryteria zatrzymania i analiza błędów	2	testowanie
19	Interpolacja i wizualizacja wyników	3	program + wykres

Internet, prawo i projekty

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
20	Rozbudowany projekt WWW	4	integracja poznanych technologii
21	Bezpieczeństwo, prawo i społeczne aspekty technologii	2	analiza przypadków
22	Projekt matematyczno-programistyczny	4	model, implementacja, testy

23	Dokumentowanie, prezentacja i ocena rozwiązania	2	wnioski i alternatywy
----	---	---	-----------------------

KLASA III – ZAKRES PODSTAWOWY

Orientacyjny wymiar: 36 godzin. W poniższej propozycji przypisano 35 godzin do tematów i bloków, a 1 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Reprezentacja informacji i urządzenia cyfrowe

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	System dwójkowy i szesnastkowy – zapis i konwersje	2	zadania obliczeniowe
2	Urządzenia cyfrowe i ich zastosowania	1	możliwości i ograniczenia

Grafika 2D i 3D

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
3	Projekt graficzny 2D – dobór techniki i narzędzi	2	raster/wektor
4	Wprowadzenie do modelowania 3D	2	podstawowe bryły i operacje
5	Projekt prostego modelu przestrzennego	2	zadanie praktyczne

Bazy danych

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
6	Podstawowe pojęcia relacyjnych baz danych	1	tabela, rekord, pole, klucz
7	Projektowanie prostej struktury bazy danych	2	tabele i relacje
8	Kwerendy, formularze i raporty	2	wyszukiwanie i prezentacja danych

Algorytmy i programowanie

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
9	Algorytmy konwersji systemów liczbowych	2	implementacja
10	Przetwarzanie tekstów	2	operacje na łańcuchach
11	Podstawy szyfrowania informacji	2	proste szyfry
12	Rekurencja – idea i przypadek bazowy	2	śledzenie wywołań
13	Proste algorytmy rekurencyjne	2	implementacja
14	Strategia zachłanna – idea i przykłady	2	kryterium wyboru

Robotyka

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
15	Wprowadzenie do sterowania układem robotycznym	2	czujniki, elementy wykonawcze
16	Programowanie zachowania robota	2	warunki i iteracja

Technologie WWW i projekt

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
17	Rozbudowa strony WWW z wykorzystaniem poznanych technologii	2	HTML/CSS oraz skrypty
18	Grafika SVG i elementy dynamiczne	1	integracja z WWW
19	Projekt internetowy lub integrujący	2	praca projektowa

KLASA III – ZAKRES ROZSZERZONY

Orientacyjny wymiar: 72 godzin. W poniższej propozycji przypisano 76 godzin do tematów i bloków, a -4 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Grafika 2D i 3D

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	Zaawansowany projekt grafiki 2D	3	łączenie technik
2	Modelowanie 3D – geometria i modyfikatory	4	np. Blender/FreeCAD/SolidWorks
3	Autorski projekt 3D	4	planowanie, wykonanie, ocena

Relacyjne bazy danych i SQL

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
4	Od problemu do modelu danych	2	analiza wymagań
5	Tabele, typy danych, klucze i relacje	2	projekt bazy
6	Podstawy języka SQL – SELECT, WHERE, ORDER BY	3	zapytania
7	Zapytania wykorzystujące wiele tabel	3	łączenie danych
8	Projekt relacyjnej bazy danych	4	projekt praktyczny

Reprezentacja liczb i metody numeryczne

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
9	Reprezentacja liczb naturalnych, całkowitych i rzeczywistych	3	ograniczenia komputerowe
10	Algorytmy konwersji reprezentacji	2	implementacja
11	Numeryczne wyznaczanie miejsc zerowych	4	porównanie metod
12	Dokładność, zbieżność i kryteria zatrzymania	3	eksperymenty

Teksty i kryptografia

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
13	Algorytmy przetwarzania tekstów	3	wyszukiwanie i transformacje
14	Szyfrowanie symetryczne – zasada działania	2	implementacja prostych metod
15	Idea kryptografii asymetrycznej	2	porównanie zastosowań
16	Projekt związany z ochroną lub przetwarzaniem informacji	2	zadanie programistyczne

Rekurencja i strategie algorytmiczne

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
17	Rekurencja w problemach liczbowych	3	analiza stosu wywołań
18	Rekurencja w grafice i fraktalach	3	wizualizacja
19	Porównanie iteracji i rekurencji	2	poprawność i efektywność
20	Algorytmy zachłanne – projektowanie kryterium wyboru	3	problemy praktyczne
21	Ograniczenia strategii zachłannej	2	kontrprzykłady

Robotyka i WWW

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
22	Projekt algorytmu sterowania robotem	3	testowanie wariantów

23	Modyfikacja i optymalizacja zachowania układu	2	diagnostyka
24	Projekt serwisu WWW integrującego poznane technologie	4	struktura, prezentacja, logika
25	Testowanie i dokumentowanie projektu internetowego	2	jakość i bezpieczeństwo

Projekt integrujący

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
26	Projekt łączący dane, programowanie i wizualizację	4	zadanie autorskie
27	Prezentacja, analiza i ocena rozwiązania	2	alternatywy i wnioski

KLASA IV – ZAKRES ROZSZERZONY

Orientacyjny wymiar: 72 godzin. W poniższej propozycji przypisano 68 godzin do tematów i bloków, a 4 godz. pozostawiono do dyspozycji nauczyciela na diagnozę, utrwalanie, ocenianie, projekty, konsultacje lub rozszerzenie wybranych zagadnień.

Grafika 2D

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
1	Projekt grafiki rastrowej – materiał użytkowy	2	np. kartka, materiał promocyjny
2	Projekt grafiki wektorowej – plakat lub identyfikacja wizualna	2	kompozycja i typografia
3	Autorski projekt 2D	2	dobór techniki i narzędzi

Modelowanie 3D

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
4	Podstawy modelowania przestrzennego – powtórzenie i uporządkowanie	2	dowolne odpowiednie środowisko
5	Zaawansowane operacje modelowania 3D	4	geometria, modyfikacja
6	Projekt indywidualnego modelu 3D	4	warianty i optymalizacja

Dokumenty i komunikacja

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
7	Rozbudowany dokument o charakterze technicznym lub naukowym	3	style, sekcje, odwołania
8	Automatyzacja struktury dokumentu i dokumentacja użytkownika	2	spisy, podpisy, szablony
9	Projekt prezentacji reklamowej produktu	2	komunikacja wizualna

Arkusz i zastosowania interdyscyplinarne

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
10	Model matematyczny w arkuszu kalkulacyjnym	2	parametry i analiza
11	Modelowanie problemu fizycznego	2	obliczenia i wykresy
12	Analiza danych historycznych lub społecznych	2	wizualizacja i interpretacja

Oprogramowanie matematyczne

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
13	Wprowadzenie do środowiska obliczeń matematycznych	2	np. SMATH Studio lub odpowiednik
14	Definiowanie zmiennych, funkcji i zależności	2	obliczenia symboliczne/liczbowe
15	Rozwiązywanie problemów matematycznych	2	model i weryfikacja
16	Modelowanie i rozwiązywanie problemów fizycznych	3	równania i wizualizacja

Gry i aplikacje

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
17	Projektowanie prostego mechanizmu gry	2	logika i interakcja
18	Realizacja prostej gry komputerowej lub mobilnej	3	projekt praktyczny
19	Aplikacje okienkowe i podstawy programowania obiektowego	3	np. Lazarus

20	Komponenty, zdarzenia i organizacja interfejsu	2	projekt aplikacji
21	Aplikacja do wizualizacji danych	3	dane + prezentacja

Pliki i dane zewnętrzne

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
22	Odczyt i zapis danych do plików	2	format danych
23	Przetwarzanie danych zewnętrznych	2	walidacja i transformacja
24	Wizualizacja danych pochodzących z pliku	2	dobór reprezentacji

Technologie internetowe i projekt końcowy

Lp.	Proponowana tematyka zajęć	Godz.	Sugestie realizacyjne
25	Projektowanie rozbudowanej strony lub aplikacji internetowej	3	HTML/CSS i poznane technologie
26	Warstwa prezentacji i logika aplikacji	2	organizacja projektu
27	Integracja poznanych technologii w projekcie końcowym	4	projekt autorski
28	Testowanie, dokumentowanie i prezentacja projektu	2	ocena i optymalizacja

Propozycje organizacji pracy projektowej

Praca projektowa może być realizowana jako osobny blok albo jako sposób podsumowania kilku wcześniej omawianych zagadnień. Wskazane jest, aby wraz ze wzrostem etapu kształcenia zwiększać samodzielność ucznia: od realizacji zadania według określonych wymagań, przez samodzielny dobór narzędzi i metod, aż do projektowania wariantów rozwiązania, testowania i uzasadniania decyzji.

- Klasa I: materiał cyfrowy łączący dokument, grafikę i prezentację; prosty program rozwiązujący problem użytkowy; mini-projekt internetowy.
- Klasa II: analiza danych w arkuszu; projekt grafiki wektorowej; program wykorzystujący funkcje i kolekcje danych; witryna HTML/CSS/JavaScript.
- Klasa III: relacyjna baza danych; model lub projekt 3D; projekt wykorzystujący rekurencję lub strategię zachłanną; prosty projekt robotyczny; serwis WWW.
- Klasa IV: projekt produktu i jego prezentacji; model 3D z dokumentacją; interdyscyplinarny model matematyczny lub fizyczny; aplikacja do wizualizacji danych; gra lub aplikacja okienkowa; projekt internetowy integrujący kilka technologii.

Uwagi dotyczące doboru narzędzi

Dobór konkretnych programów i środowisk pozostaje decyzją nauczyciela. Możliwe jest wykorzystanie zarówno oprogramowania instalowanego lokalnie, jak i narzędzi internetowych, pod warunkiem zapewnienia zgodności z celami zajęć, bezpieczeństwa uczniów i możliwości technicznych szkoły. W przypadku programowania wskazane jest utrzymywanie ciągłości pracy z językami używanymi w danej szkole, przy jednoczesnym eksponowaniu uniwersalnych pojęć algorytmicznych i programistycznych.

Narzędzia generatywnej sztucznej inteligencji mogą być wykorzystywane pomocniczo w procesie uczenia się, analizowania rozwiązań i pracy projektowej, jeżeli nauczyciel na to zezwala. Uczeń powinien być w stanie wyjaśnić, zmodyfikować i zastosować rozwiązanie, które przedstawia jako rezultat swojej pracy.