

Ocenę dopuszczającą uzyskuje uczeń, który:

- Zna ogólną budowę programu i najważniejsze elementy języka – słowa kluczowe, instrukcje, wyrażenia, zasady składni.
- Potrafi zrealizować prosty algorytm liniowy w języku wysokiego poziomu; potrafi skompilować i uruchomić gotowy program.
- Zna podstawowe typy strukturalne – tablice i pliki i potrafi je zadeklarować.
- Potrafi omówić wybrany algorytm porządkowania elementów (np. metodą przez wybór).
- Zna sposoby reprezentacji danych liczbowych i nieliczbowych.
- Zna główne cechy pozycyjnej reprezentacji liczb.
- Potrafi utworzyć proste formularze w edytorze tekstu.
- Wymienia przykładowe programy do projektowania i tworzenia stron internetowych.
- Potrafi wymienić podstawowe elementy, z których składa się strona WWW.
- W stopniu podstawowym posługuje się wybranym programem do tworzenia stron.
- Tworzy nieskomplikowaną stronę, na której: wstawia tytuł, formatuje tekst, umieszcza obraz.

Ocenę dostateczną uzyskuje uczeń, który:

- Spełnia kryteria niższej oceny.
- Zapisuje program w czytelnej postaci – stosuje wcięcia, komentarze.
- Rozumie pojęcia: implementacja, kompilacja, uruchomienie, testowanie.
- Rozumie znaczenie i działanie podstawowych instrukcji wybranego języka programowania wysokiego poziomu.
- Rozróżnia i poprawia błędy kompilacji i błędy wykonania.
- Potrafi zrealizować algorytmy iteracyjne w języku wysokiego poziomu.
- Zna podstawowe zasady poprawnego programowania; testuje tworzone programy; wie, jak unikać problemów, takich jak np. zapętlenie się programu.
- Zna podstawowe algorytmy sortowania tablic.
- Potrafi zamienić całkowitą liczbę dziesiętną na liczbę w innych systemach.
- Wie, co to jest system binarny, i potrafi dokonać zamiany liczby z systemu dziesiętnego na binarny i odwrotnie.
- Potrafi utworzyć formularze w edytorze tekstu i arkusza kalkulacyjnym.
- Wie, co to jest język znaczników HTML, i potrafi omówić strukturę pliku w tym języku.
- Zna podstawy języka znaczników HTML i potrafi wykonać prostą stronę na zadany przez nauczyciela temat.
- Z pomocą nauczyciela projektuje wygląd strony.
- Planuje jej zawartość (teksty, rysunki, dźwięki, animacje) i umieszcza na niej ww. elementy.

Ocenę dobrą uzyskuje uczeń, który:

- Spełnia kryteria niższej oceny.
- Określa problemy, w których występuje rekurencja i podaje przykłady „zjawisk rekurencyjnych” – wziętych z życia i zadań szkolnych.
- Zna rekurencyjną realizację wybranego algorytmu, np. silni.
- Rozumie, co to jest złożoność algorytmu i potrafi określić liczbę operacji wykonywanych na elementach zbioru w wybranym algorytmie sortowania.

- Potrafi prezentować złożone algorytmy (z podprogramami) w wybranym języku programowania.
- Zna rekurencyjne realizacje prostych algorytmów.
- Rozumie i stosuje zasady programowania strukturalnego.
- Wie, na czym polega różnica pomiędzy przekazywaniem parametrów przez zmienną i przez wartość w procedurach.
- Wie, jakie znaczenie ma zasięg działania zmiennej.
- Rozumie zasady postępowania przy rozwiązywaniu problemu metodą zstępującą.
- Zna zasady działania wybranych algorytmów sortowania.
- Potrafi zastosować łańcuchowy i tablicowy typ danych w zadaniach.
- Potrafi wykonać dowolną konwersję pomiędzy systemem dziesiętnym, dwójkowym, ósemkowym i szesnastkowym.
- Zna sposób zapisu liczby całkowitej i rzeczywistej (zmiennoprzecinkowej).
- Potrafi samodzielnie tworzyć formularze i zapisywać je, jako szablony.
- Potrafi samodzielnie zaprojektować wygląd strony WWW.
- Zna reguły poprawnego projektowania układu strony, m.in. dba o jej czytelność i przejrzystość, o poprawność redakcyjną i merytoryczną oraz prawną umieszczanych na niej tekstów i materiałów.
- Zna zaawansowane możliwości języka HTML: tabele, ramki, style.
- Zna sposoby publikowania stron w Internecie oraz wady i zalety tych sposobów.

Ocenę bardzo dobrą uzyskuje uczeń, który:

- Zna metodę „dziel i zwyciężaj”, algorytm generowania liczb Fibonacciego, schemat Hornera.
- Omawia ich iteracyjną realizację i potrafi przedstawić jeden z nich w wybranej notacji.
- Zna przynajmniej jeden algorytm numeryczny, np. obliczanie wartości pierwiastka kwadratowego
- Wskazuje różnicę między rekurencją a iteracją.
- Zna rekurencyjną realizację wybranych algorytmów, np. silnie i algorytm Euklidesa.
- Potrafi zamienić algorytm zapisany iteracyjnie na postać rekurencyjną.
- Potrafi porównać złożoność różnych algorytmów tego samego zadania dla tych samych danych.
- Wie, kiedy algorytm jest uniwersalny.
- Zna inne algorytmy sortowania, np. przez wstawianie i przez indeksowanie.
- Samodzielnie potrafi zapisać procedury sortujące w wybranym języku programowania wysokiego poziomu.
- Wie, na czym polega różnica pomiędzy przekazywaniem parametrów przez zmienną i przez wartość w procedurach.
- Wie, jakie znaczenie ma zasięg działania zmiennej.
- Potrafi porównać złożoność różnych algorytmów tego samego zadania dla tych samych danych.
- Wie, kiedy algorytm jest uniwersalny.
- Samodzielnie korzysta z wybranego programu do tworzenia stron.
- Zna operacje logiczne na liczbach binarnych i przesunięcia bitowe.
- Potrafi zapisać w języku programowania wysokiego poziomu algorytm konwersji liczb z dowolnego systemu pozycyjnego na inny.
- Wykonuje sprawnie operacje na liczbach zapisanych w różnych systemach pozycyjnych.
- Potrafi przekazać dane z formularzy do baz danych.
- Potrafi wykorzystać nowo poznane funkcje języka HTML. Wykorzystuje je do udoskonalenia istniejących już, swoich własnych stron.

- Włącza licznik odwiedzin na stronie. Dodaje inne typowe elementy: forum, księgę gości.
- Potrafi wstawić gotowe skrypty dla osiągnięcia nieskomplikowanych efektów wizualnych na stronie.
- Potrafi utworzyć formularz na stronie WWW.
- Potrafi opublikować stronę w Internecie.

Ocenę celującą uzyskuje uczeń, który:

- Spełnia kryteria niższej oceny
- Rozumie dokładnie technikę rekurencji (znaczenie stosu).
- Potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję.
- Zna trudniejsze algorytmy, np. wieże Hanoi, problem ośmiu hetmanów.
- Zna inne algorytmy numeryczne, np. wyznaczanie miejsca zerowego funkcji.
- Korzysta samodzielnie z dodatkowej literatury.
- Ocenia złożoność czasową i pamięciową algorytmu. Zna odpowiednie wzory.
- Ocenia efektywność działania programu.
- Bierze udział w konkursach i olimpiadach.
- Zna trudniejsze algorytmy, np. wieże Hanoi, problem ośmiu hetmanów.
- Zna inne techniki sortowania, np. sortowanie przez scalanie ciągów. Potrafi zapisać je w różnych notacjach (również w języku programowania wysokiego poziomu).
- Korzysta samodzielnie z dodatkowej literatury.
- Ocenia złożoność czasową i pamięciową algorytmu. Zna odpowiednie wzory.
- Ocenia efektywność działania programu.
- Wie, na czym polega programowanie obiektowe i zdarzeniowe.
- Potrafi stosować techniki programowania dynamicznego lub programowania obiektowego.
- Zna i rozumie podobieństwa i różnice w strukturze programu zapisanego w różnych językach programowania.
- Potrafi zastosować zmienne typu wskaźnikowego w zadaniach.
- Zna struktury listowe, np. stos, kolejkę, listę.
- Rozumie i potrafi zastosować typ obiektowy.
- Bierze udział w konkursach i olimpiadach.

¹ Opracowano na podstawie „Przykładów wymagań na poszczególne oceny szkolne” Grażyny Koby