

## Przedmiotowy system oceniania

Zawód: Technik Informatyk

Nr programu: 312[ 01] /T,SP/MENiS/ 2004.06.14

Przedmiot: Programowanie Strukturalne i Obiektowe

Klasa: druga

Dział	Dopuszczający	Dostateczny	Dobry	Bardzo dobry	Celujący
<b>Algorytmika i pseudoprogramowanie</b>	<ul style="list-style-type: none"> <li>zna podstawowe definicje co to jest algorytm i program komputerowy</li> <li>zna sposoby zapisu algorytmu</li> <li>zna bloki algorytmu zapisane w postaci graficznej</li> <li>potrafi odróżnić algorytm liniowy od algorytmu z warunkiem</li> <li>wie co to iteracja</li> <li>potrafi posłużyć programem edukacyjnym ELI do prezentacji algorytmów</li> <li>wie, że można zaprezentować działanie niektórych algorytmów w arkuszu kalkulacyjnym</li> <li>wie co to pseudojęzyk programowania,</li> <li>wie, że można zapisać algorytm pseudojęzyku programowania,</li> <li>wie do czego służy środowisko Algo</li> </ul>	<ul style="list-style-type: none"> <li>wie jaki jest związek między problemem a algorytmem,</li> <li>zna sposoby zapisu algorytmów i potrafi je krótko opisać,</li> <li>potrafi przedstawić algorytm w postaci graficznej</li> <li>zna specyfikację algorytmu,</li> <li>potrafi zapisać algorytm jako listę kroków,</li> <li>zna różnice między algorytmem liniowym a z warunkiem,</li> <li>wie jak wygląda algorytm z warunkiem zagnieżdżonym,</li> <li>potrafi przedstawić algorytm w postaci drzewa decyzyjnego,</li> <li>potrafi wykorzystać program Eliz do sprawdzenia poprawności algorytmu</li> <li>potrafi zaprezentować prosty algorytm w arkuszu kalkulacyjnym</li> <li>potrafi zapisać algorytm w pseudojęzyku programowania,</li> <li>zna środowisko Algo</li> </ul>	<ul style="list-style-type: none"> <li>potrafi dobrze zdefiniować związek pomiędzy algorytmem a problemem, podaje przykłady,</li> <li>zna wszystkie sposoby zapisu algorytmów i potrafi je opisać,</li> <li>potrafi zapisać algorytm w formie opisowej, graficznej, jako lista kroków i jako drzewo decyzyjne,</li> <li>zna podział algorytmów ze względu na ich strukturę,</li> <li>samodzielnie realizuje wcześniej poznane algorytmy w programie Eli(np. rozwiązanie równania kwadratowego, liniowego, wartość bezwzględna)</li> <li>potrafi samodzielnie zaprezentować poznane algorytmy w arkuszu kalkulacyjnym,</li> <li>wie na czym polega pseudoprogramowanie,</li> <li>dobrze zna środowisko Algo</li> <li>potrafi zapisać każdy algorytm w pseudojęzyku programowania,</li> </ul>	<ul style="list-style-type: none"> <li>bardzo dobrze zna pojęcia i definicje związane z algorytmiką i pseudoprogramowaniem</li> <li>bezproblemowo posługuje się wszystkimi metodami zapisu algorytmu,</li> <li>bezproblemowo prezentuje poznane algorytmy przy użyciu programu Eli i arkusza kalkulacyjnego,</li> <li>pisze programy w pseudojęzyku programowania</li> </ul>	<p>Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania, a ponadto spełnia jeden z podpunktów:</p> <ul style="list-style-type: none"> <li>świetnie potrafi przedstawić zadany problem w postaci algorytmu (w dowolnej postaci graficznej, listy kroków itd.)</li> <li>zna przynajmniej jeden język programowania i potrafi zrealizować poznane algorytmy przy jego pomocy</li> </ul>

<b>Wprowadzenie do programowania strukturalnego Pascala</b>	<ul style="list-style-type: none"> <li>• wie, że są różne języki programowania,</li> <li>• zna podstawowe pojęcia związane z programowaniem: program komputerowy, języki programowania, kompilator, interpretator, konsolidator,</li> <li>• zna etapy tworzenia programu,</li> <li>• zna edytor środowiska Pascal,</li> <li>• zna podstawowe słowa kluczowe w języku Pascal,</li> <li>• zna podstawowe typy danych w języku Pascal (całkowity, rzeczywisty, znakowy, łańcuchowy, logiczny),</li> <li>• wie, że są instrukcje wyjścia/wejścia,</li> <li>• wie, że są funkcje i procedury,</li> <li>• wie, że są instrukcje warunkowe, iteracyjne i złożone w programach,</li> <li>• wie, że można realizować wcześniej poznane algorytmy przy pomocy języka Pascal</li> </ul>	<ul style="list-style-type: none"> <li>• potrafi wymienić kilka języków programowania,</li> <li>• zna podstawowe pojęcia związane z programowaniem: program komputerowy, języki programowania, kompilator, interpretator, konsolidator,</li> <li>• zna etapy tworzenia programu,</li> <li>• wie na czym polega programowanie strukturalne,</li> <li>• zna edytor środowiska Pascal,</li> <li>• zna podstawowe typy danych w języku Pascal (całkowity, rzeczywisty, znakowy, łańcuchowy, logiczny) i wie kiedy je stosować,</li> <li>• wie, że są funkcje i procedury i wie kiedy je stosować,</li> <li>• wie, że są instrukcje warunkowe, iteracyjne i złożone w programach i wie kiedy je stosować,</li> <li>• wie, że można realizować wcześniej poznane algorytmy iteracyjne i z warunkiem przy pomocy języka Pascal</li> </ul>	<ul style="list-style-type: none"> <li>• zna klasyfikację języków programowania, potrafi wymienić przykłady zastosowania poszczególnych języków programowania,</li> <li>• zna i rozumie pojęcia związane z programowaniem</li> <li>• zna i rozumie etapy tworzenia programu</li> <li>• wie na czym polega programowanie strukturalne, potrafi wymienić języki programowania strukturalnego,</li> <li>• dobrze zna edytor środowiska Pascal,</li> <li>• zna wszystkie typy danych w języku Pascal, odpowiednio je stosuje w trakcie pisania programów,</li> <li>• potrafi programować z użyciem funkcji i procedur,</li> <li>• potrafi programować wykorzystując instrukcje warunkowe i iteracyjne</li> </ul>	<ul style="list-style-type: none"> <li>• biegle klasyfikuje języki programowania,</li> <li>• bezproblemowo potrafi napisać program w języku Pascal z zgodnie z wcześniej poznanymi algorytmami,</li> <li>• bezproblemowo pisze programy świadomie stosując odpowiednie typy danych,</li> <li>• biegle programuje stosując funkcje i procedury,</li> <li>• bezproblemowo wykorzystuje w trakcie programowania instrukcje warunkowe i iteracyjne,</li> <li>• potrafi zrealizować (napisać program) na dowolnie zadany problem</li> </ul>	<p>Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania, a ponadto spełnia jeden z podpunktów:</p> <ul style="list-style-type: none"> <li>• zna przeznaczenie i zastosowanie innych języków programowania nie omawianych na lekcjach</li> <li>• świetnie programuje w Pascalu, zna inne języki programowania i potrafi w nich napisać program</li> <li>• zna różnice pomiędzy programowaniem obiektowym a strukturalnym</li> </ul>
---	--	--	--	--	--

**Zawód: Technik Informatyk****Nr programu:** 312[ 01] /T,SP/MENiS/ 2004.06.14**Przedmiot: Programowanie Strukturalne i Obiektowe****Klasa:** trzecia

Dział	Dopuszczający	Dostateczny	Dobry	Bardzo dobry	Celujący
<b>Podprogramy</b>	<ul style="list-style-type: none"> <li>• zapisuje w postaci programu algorytmy iteracyjne, w których nie jest z góry określona liczba kroków iteracji</li> <li>• wie, jak zadeklarować i wywołać funkcję w Turbo Pascalu;</li> <li>• potrafi przejść do trybu graficznego w Turbo Pascalu;</li> <li>• omawia i analizuje wybrane algorytmy sortowania (przez wybór, bąbelkowy, kubełkowy) przygotowane np. w postaci gotowych schematów blokowych lub programów komputerowych</li> <li>• wyjaśnia różnice między iteracją a rekurencją</li> </ul>	<ul style="list-style-type: none"> <li>• definiuje prostą funkcję z jednym parametrem</li> <li>• omawia iteracyjną realizację algorytmów: Euklidesa, generowania liczb Fibonacciego, schemat Hornera;</li> <li>• wyjaśnia metodę „dziel i zwyciężaj” na przykładzie algorytmu znajdowania minimum i maksimum równocześnie</li> <li>• stosuje w programach podstawowe procedury graficzne, korzystając z podręcznika (rysuje punkty, proste i krzywe, wypełnia barwą wyznaczony obszar)</li> <li>• wie, na czym polega rekurencja i podaje przykłady zjawisk rekurencyjnych;</li> </ul>	<ul style="list-style-type: none"> <li>• zna działanie instrukcji iteracyjnych while i repeat;</li> <li>• wyjaśnia, kiedy stosujemy funkcje, a kiedy procedury</li> <li>• potrafi zapisać w języku Turbo Pascal iteracyjne wersje algorytmów,</li> <li>• stosuje procedury graficzne w realizacji złożonych programów, np. animacji;</li> <li>• potrafi zapisać wybrane algorytmy sortowania w postaci programów w języku Turbo Pascal</li> <li>• potrafi zapisać algorytm rekurencyjny w postaci programu</li> </ul>	<ul style="list-style-type: none"> <li>• samodzielnie dobiera odpowiednią instrukcję iteracyjną, zależnie od warunków zadania</li> <li>• pisze trudniejsze programy, definiując w nich odpowiednio funkcje i/lub procedury;</li> <li>• potrafi zapisać w języku Turbo Pascal iteracyjne wersje algorytmów: Euklidesa, generowania liczb Fibonacciego, schemat Hornera i algorytm znajdowania minimum i maksimum równocześnie</li> <li>• potrafi napisać program rysujący wykres funkcji, np. <math>y=\sin(x)</math>, spiralę Archimedesesa</li> <li>• potrafi zapisać wybrane algorytmy w postaci programów w języku Turbo Pascal w wersji rekurencyjnej, swobodnie zamienia wersję iteracyjną na rekurencyjną</li> </ul>	Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania.

<b>Typy strukturalne</b>	<ul style="list-style-type: none"> <li>• wie czym jest tablica – jako przykład typu strukturalnego</li> <li>• zna tablice jedno i dwu wymiarowe</li> <li>• zna procedury do wczytywania i wyprowadzania elementów tablicy</li> <li>• potrafi wymienić klasyczne algorytmy działające na tablicach</li> <li>• potrafi zapisać liczby w różnych systemach liczbowych</li> <li>• wie do czego służy Sito Eratostenesa</li> <li>• zna sposoby sortowania tablicy (bąbelkowe, szybkie, przez wstawianie, przez wybór)</li> <li>• zna szyfr cezara</li> </ul>	<ul style="list-style-type: none"> <li>• potrafi przedstawić tablicę jako przykład typu strukturalnego</li> <li>• wie czym jest typ strukturalny</li> <li>• potrafi zastosować procedury do wczytywania i wyprowadzania elementów tablicy</li> <li>• potrafi zastosować algorytmy działające na tablicach</li> <li>• zna metody sortowania tablicy i umie je zastosować (bąbelkowe, szybkie, przez wstawianie, przez wybór)</li> <li>• wykorzystuje tablice tekstowe</li> <li>• potrafi wyszukać i policzyć znaki w tekście</li> </ul>	<ul style="list-style-type: none"> <li>• dobrze rozumie zagadnienia tablic jako przykładu typu strukturalnego</li> <li>• potrafi omówić typ strukturalny i wie kiedy się go stosuje</li> <li>• zna dobrze klasyczne algorytmy działające na tablicach i potrafi je wykorzystać radzi sobie z wyszukiwaniem elementu maksymalnego (minimalnego) w tablicy.</li> <li>• zapisuje liczby w różnych systemach liczbowych</li> <li>• radzi sobie z tablicami tekstowymi (Czy dany ciąg jest palindromem)</li> </ul>	<ul style="list-style-type: none"> <li>• bardzo dobrze rozumie zagadnienia tablic</li> <li>• wykorzystuje w programach tablice i algorytmy przeszukiwania i sortowania elementów tablicy</li> </ul>	<p>Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania, a ponadto spełnia jeden z podpunktów:</p> <ul style="list-style-type: none"> <li>• zna inne niż poznane na zajęciach metody przeszukiwania tablic</li> </ul>
--------------------------	---	--	---	---	---

<p style="text-align: center;"><b>Dynamiczne struktury danych</b></p>	<ul style="list-style-type: none"> <li>• zna typ wskaźnikowy i jego zastosowanie</li> <li>• wie jak zadeklarować zmienne wskaźnikowe</li> <li>• potrafi wykonać podstawowe operacje na wskaźnikach</li> <li>• wie, że można wykorzystać wskaźniki do przeszukiwania tablic</li> <li>• zna stos i kolejkę jako przykład struktury danych</li> <li>• wie, że można tworzyć i przetwarzać dynamiczne struktury danych takie jak (listy drzewa grafy)</li> </ul>	<ul style="list-style-type: none"> <li>• potrafi zdefiniować typ wskaźnikowy</li> <li>• deklaruje zmienne wskaźnikowe</li> <li>• wykonuje podstawowe operacje na wskaźnikach</li> <li>• potrafi zastosować wskaźniki w tablicach</li> <li>• wykorzystuje stos i kolejkę (jako struktury danych)</li> <li>• próbuje tworzyć i przetwarzać dynamiczne struktury danych takie jak: listy, drzewa, grafy</li> <li>• wykonuje losowanie danych</li> </ul>	<ul style="list-style-type: none"> <li>• rozumie i poprawnie deklaruje zmienne wskaźnikowe</li> <li>• poprawnie wykonuje operacje na wskaźnikach</li> <li>• dobrze stosuje wskaźniki w tablicach</li> <li>• potrafi rozwiązać zadanie z wykorzystaniem wskaźników</li> <li>• stosuje stos – odwrotna notacja polska(ONP)</li> <li>• potrafi losować dane i sprawdzać ich poprawność</li> </ul>	<ul style="list-style-type: none"> <li>• bardzo dobrze wie jak wykorzystać poprawnie wskaźniki w odniesieniu do tablic</li> <li>• zna i stosuje dynamiczne struktury danych (stos, kolejka)</li> <li>• bardzo dobrze wykorzystuje tworzenie i przetwarzanie dynamicznych struktur danych takich jak listy, drzewa, grafy</li> <li>• umiejętnie stosuje stos – odwrotna notacja polska(ONP)</li> <li>• bezbłędnie potrafi losować dane i sprawdzać ich poprawność</li> </ul>	<p>Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania.</p>
<p style="text-align: center;"><b>Elementy zaawansowanego programowania w Pascalu 18</b></p>	<ul style="list-style-type: none"> <li>• zna obsługę kompilatora</li> <li>• wie, że można programować urządzenie peryferyjne</li> <li>• zna podstawowe instrukcje służące do komunikacji z urządzeniami peryferyjnymi</li> <li>• wie czym są rejestry procesora</li> <li>• wie do czego służy debugger</li> <li>• zna definicję złożoności obliczeniowej algorytmu</li> <li>• wie na czym polega optymalizacja programów</li> </ul>	<ul style="list-style-type: none"> <li>• poprawnie posługuje się kompilatorem</li> <li>• potrafi napisać prosty program komunikujący się z urządzeniami peryferyjnymi</li> <li>• zna rejestry procesora</li> <li>• potrafi wykorzystać debugger do wykrycia błędów</li> <li>• wie jak wykonać dokumentację programu</li> </ul>	<ul style="list-style-type: none"> <li>• zna dyrektywy kompilatora.</li> <li>• potrafi programować obsługę urządzeń peryferyjnych (drukarka, skaner)</li> <li>• wykazuje się dobrą znajomością rejestrów procesora</li> <li>• potrafi wykorzystać debugger do wykrywania i usuwania błędów w programach</li> <li>• wie na czym polega złożoności obliczeniowa algorytmów</li> <li>• potrafi zoptymalizować działanie programów</li> <li>• bezproblemowo radzi sobie z wykonaniem dokumentacji programu</li> </ul>	<ul style="list-style-type: none"> <li>• świetnie opanował dyrektywy kompilatora.</li> <li>• potrafi programować obsługę urządzeń peryferyjnych (drukarka, skaner)</li> <li>• bardzo dobra znajomość rejestrów procesora</li> <li>• bezproblemowo posługuje się debuggerem do wykrywania i usuwania błędów w programach</li> <li>• rozumie istotę złożoności obliczeniowej i pamięciowej algorytmów</li> <li>• potrafi zoptymalizować działanie programów</li> <li>• bezproblemowo radzi sobie z wykonaniem dokumentacji programu</li> </ul>	<p>Ocenę tę otrzymuje słuchacz, którego wiedza wykracza poza obowiązujący program nauczania.</p>