WYŻSZA SZKOŁA GOSPODARKI W BYDGOSZCZY

KIERUNEK : **MECHATRONIKA**

Przedmiot : PROGRAMOWANIE (STRUKTURALNE I OBIEKTOWE)

Prowadzący: INŻ. MARCIN JASIŃSKI

SPRAWOZDANIE Z ĆWICZEŃ

- PROJEKTOWYCH/LABORATORYJNYCH \*

Nr …4..

**1) Budowa i zastosowanie pętli: „for”, ”while„ oraz „do…while”.**

**2) Konstrukcja i tworzenie tablic.**

**3) Operacje wejścia – wyjścia, metoda typu „scaner”**

**Budowa pętli for i jej przykład.**

*for (wyrażenie początkowe; wyrażenie warunkowe; wyrażenie modyfikujące){*

*//instrukcja do wykonania*

*}*

*np.*

*Class Main*

*{*

*public static void main (String args[]) {*

*for (int I = 1; I <= 10; i++){*

*System.out.println (“Java”);*

Z powyższego zapisu wynika, że zmienna „i” jest tak zwaną zmienną iteracyjną, która na początku ma wartość 1. Następnie każdorazowo po wykonaniu pętli (iteracyjnie)jest ona zwiększana o 1 (++i) aż do momentu spełnienia warunku (i <= 10), przy czym każdorazowo również wykonywana jest instrukcja. *System.out.println (“Java”);*

Pętla “for” może również przyjmować inną postać:

*wyrażenie początkowe;*

*class Main*

*{*

*public static void main (String args[]) {*

*int i = 1;*

*for (; i <= 10;){*

*System.out.println ("Java");*

*i++;*

*}*

*for (; wyrażenie warunkowe;){*

*// instrukcje do wykonania*

*wyrażenie modyfikujące*

*}*

Przy korzystaniu z takiej formy zapisu należy pamiętać, że do prawidłowego zapisu potrzebne są dwa średniki wewnątrz okrągłych nawiasów.

**Budowa pętli „ while „**

Różnica między pętlą for i pętlą while polega na tym, że używając pętli for wiemy ile operacji do wykonania będziemy mieli, gdyż jest ona zawarta w samej zmiennej. Natomiast, w pętli while ilość operacji do wykonania wynika z wartości zwróconej przez funkcję.

Schemat pętli while

*while (wyrażenie warunkowe) {*

*//instrukcje*

Np.

*class Main {*

*public static void main (String args[]) {*

*int i = 1;*

*while (i <= 10){*

*System.out.println ("Java");*

*i++;*

Z przykładu tego wynika, że pętla zadziała w sposób iteracyjny, aż do spełnienia wyrażenia warunkowego(i<=10).

**Pętla do…while.**

**Budowa pętli do…while.**

*do{*

*// instrukcje;*

*}*

*while (warunek);*

Jeszcze jednym typem pętli jest pętla do…while, która w swojej konstrukcji jest bardzo podobna do pętli while, jednak zasadniczą różnicą jest fakt, iż w pętli do…while najpierw wykonywana jest instrukcja a dopiero później sprawdzany jest warunek. Zobrazujemy tą sytuację na podstawie przykładu.

*class Main {*

*public static void main (String args[]) {*

*int i = 10;*

*do{*

*System.out.println ("Java");*

*}*

*while (i++ <= 9);*

Widzimy, że warunek zakończenia pętli (i++ <= 9) jest sprzeczny z wartością przepisaną dla zmiennej „i”.

Jednak po uruchomieniu naszego programu na ekranie pojawi się jeden raz napis Java.

Czyli najpierw została wykonana instrukcja następnie sprawdzony został warunek.

**Wnioski.**

Omawiane pętle odgrywają bardzo dużą rolę w tworzeniu programu, gdyż pozwalają na iteracyjne sprawdzenie warunków i dobranie odpowiednich instrukcji w zależności od ich zgodności lub fałszu z założeniem warunkowym.

**Tworzenie tablic.**

Schemat tworzenia tablic.

*typ\_tablicy nazwa\_tablicy[] = new typ\_tablicy[liczba\_elementów];*

*np.*

*int tablica[] = new int[3]*

*O rodzaju danych w tablicy decydujemy w chwili jej tworzenia .W naszym przypadku stworzyliśmy tablicę zawierającą wartości całkowite zgodne z typem int.*

*Należy pamiętać, że numeracja zawsze zaczyna się od indeksu „0”*

*Istnieje również możliwość jednoczesnego utworzenia tablicy z przypisaniem dla poszczególnych komórek określonych wartości.*

*typ\_tablicy nazwa\_tablicy[] = {wartość1, wartość2, ... , wartośćn}*

*Wielkość tablicy w takim przypadku wynika z ilości elementów zawartych w { , , }*

*Przykład .*

*int tablica[] = {1, 2, 25, 4};*

*Zapis ten oznacza, że dla zmiennej o nazwie „tablica” utworzyliśmy czteroelementową tablicę zawierającą w kolejnych komórkach liczby*

*1 ,2 ,25 ,4*

*W celu uzyskania dostępu do wartości zapisanej w danej komórce musimy wywołać jej indeks w nawiasach [], pamiętając, że pierwsza komórka ma indeks 0 a nie 1.*

*Zatem:*

*int tablica[] = {1, 8, 15, 2};*

*System.out.println("tablica[0] = " + tablica[0]);*

*System.out.println("tablica[1] = " + tablica[1]);*

*System.out.println("tablica[2] = " + tablica[2]);*

*System.out.println("tablica[3] = " + tablica[3]);*

Otrzymamy zapis

*tablica[0] = 1*

*tablica[1] = 8*

*tablica[2] = 15*

*tablica[3] = 2*

Zawartość komórek może być wielokrotnie zapisywana i odczytywana jednak należy pamięta, że komórka zawsze pamięta ostatni zapis. Przedstawmy to na przykładzie;

*public static void main (String args[]) {*

*int tablica[] = new int[2];*

*tablica[0] = 20;*

*tablica[1] = 40;*

*System.out.println("tablica[0] = " + tablica[0]);*

*System.out.println("tablica[1] = " + tablica[1])*

*tablica[0] = 39;*

*tablica[1] = 40;*

*System.out.println("tablica[0] = " + tablica[0]);*

*System.out.println("tablica[1] = " + tablica[1]);*

Pierwotnie zostaną wyświetlone wartości 20 i 40, następnie 39 i 40.

Do odczytu rozmiaru tablicy używamy właściwości **tab.lenght,** która określa liczbę komórek.

Przykład.

*class Main {*

*public static void main (String args[]) {*

*int tab[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,*

*10, 9, 8, 7, 6, 5, 4, 3, 2, 1*

*};*

*System.out.println("Wielkość tablicy: " + tab.length)*

***WNIOSKI****.*

*Za pomocą tablic możemy przedstawić w sposób uporządkowany strukturę danych określonego typu. Zapisywać i zmieniać wartości komórek tworzących strukturę tabeli a także odwoływać się do nich.*

***Pobieranie, czytanie danych przy pomocy skanera.***

Dzięki operacjom wejścia –wyjścia możemy wprowadzać dane z klawiatury, czy dokonywać operacji na plikach. W Javie tego typu działania opierają się na strumieniach wejściowych (System.in) dzięki, którym możemy odczytywać dane i strumieniach wyjściowych (System.out) do ich zapisywania.

Klasa Scanner z pakietu java.util należy do strumienia wejściowego i wywołujemy ją za pomocą konstrukcji w postaci:

Scanner nazwa = new Scanner(System.in);

String x = nazwa.next(); // deklarowanie zmiennej

nazwa.close(); // zamykamy scanner

OCENA:……………

WYKONAŁ : Mariusz Maksoń SPRAWDZIŁ:

…………………… ………………………..

mgr K. Czarnecki

/Imię i nazwisko nr albumu/

\*niepotrzebne skreślić