

Wstęp do programowania w języku C (kolokwium 2018)

1. (50p) Punkt na płaszczyźnie może być reprezentowany przez parę liczb (x, y) określających jego współrzędne w ustalonym układzie współrzędnych. Odcinek jest określony przez parę punktów, które są jego końcami, a prostą o równaniu $Ax + By + C = 0$ może reprezentować trójka liczb (A, B, C) . Napisać program w języku C, który zawiera definicje struktur reprezentujących punkt, odcinek i prostą oraz funkcje:
- (a) Dla zadanych jako parametry dwóch punktów A i B funkcja *prosta*, która wyznacza i zwraca prostą przechodzącą przez punkty A i B .
 - (b) Dla zadanych jako parametry prostej p i punktu A funkcja *prostopadla*, która wyznacza i zwraca prostą prostopadłą do prostej p i przechodzącą przez A .
 - (c) Dla zadanych jako parametry dwóch prostych p i q funkcja *przeciecie*, która wyznacza i zwraca punkt ich przecięcia.
 - (d) Dla zadanych jako parametry odcinka \overline{AB} i punktu C funkcja *odleglosc*, która oblicza i zwraca odległość C od najbliższego punktu z odcinka \overline{AB} . Tym najbliższym punktem jest albo jeden z końców odcinka, albo punkt przecięcia prostej przechodzącej przez C i prostopadłej do odcinka \overline{AB} .

Program główny powinien wczytać n oraz sekwencję trójek punktów A_i, B_i, C_i , $i = 1, \dots, n$ i dla każdej trójki wypisać odległość punktu C_i od odcinka $\overline{A_i B_i}$.

2. (50p) Węzeł drzewa binarnego zawiera następujące informacje: (1) priorytet (typu int), (2) wartość (typu double), (3) wskaźniki na lewe i prawe poddrzewa oraz (4) sumę liczb węzłów w prawym i lewym poddrzewie. Zakładamy, że w drzewie węzły są uporządkowane zgodnie z porządkiem kopcowym priorytetów, tzn. priorytet w każdym węźle jest nie mniejszy niż priorytety w jego potomkach. Drzewo (zwane w tym przypadku kopcem) jest reprezentowane przez wskaźnik na korzeń (jeśli kopiec zawiera co najmniej jeden element) lub przez NULL (w przeciwnym przypadku). Zdefiniować moduł KOPIEC w języku C (tzn. podać zawartość plików *kopiec.h* oraz *kopiec.c*), w którym:
- (a) Za pomocą `typedef` zdefiniowany jest typ wskaźnikowy KOPIEC.
 - (b) Zdefiniowana jest funkcja *inicjuj*, która zwraca wartość typu KOPIEC. Kopiec początkowo powinien być pusty.
 - (c) Zdefiniowana jest funkcja *dodaj*, która dla zadanych jako parametry: kopca *kopiec* (typu KOPIEC) oraz danych *priorytet* i *wartosc* wstawi tę parę do struktury *kopiec* zachowując porządek kopcowy i dodatkowo przy wyborze poddrzewa, do którego należy wstawiać wybierze to z nich, które zawiera mniej węzłów (jeśli zawierają tyle samo, to wstawić należy do lewego poddrzewa).
 - (d) Zdefiniowana jest funkcja *usunmax* która dla zadanego jako parametr kopca *kopiec* zwróci wartość w korzeniu kopca oraz usunie ten korzeń z kopca zachowując porządek kopcowy.

3. (50p) Napisać funkcję w języku C, która rozwiązuje następujący problem plecakowy: Danych jest N przedmiotów, z których każdy opisany jest poprzez dwie liczby całkowite dodatnie: objętość W_i oraz wartość V_i , $i = 1, \dots, N$. Dana jest również objętość plecaka $Wmax \leq 1000$ (typu int). Zadaniem funkcji jest wybrać podzbiór przedmiotów i_1, i_2, \dots, i_k (każdy przedmiot może być wybrany tylko raz) taki że suma ich objętości $\sum_{j=1}^k W_{i_j}$ jest nie większa niż $Wmax$ oraz suma wartości $\sum_{j=1}^k V_{i_j}$ jest jak największa. Funkcja powinna zwrócić taką maksymalną sumę wartości.

Wskazówka: Najprostszą metodą rozwiązania tego problemu jest wypełnienie tablicy prostokątnej T o rozmiarach $(Wmax + 1) \times (N + 1)$, gdzie zerowy wiersz i zerowa kolumna wypełnione są zerami, a element $T[w, i]$ z kolejnego wiersza w jest równy albo $T[w, i - 1]$ jeśli $W_i > w$, albo $\max(T[w, i - 1], T[w - W_i, i - 1] + V_i)$, w przeciwnym przypadku. Funkcja powinna zwrócić wartość $T[Wmax, N]$.