

Zadanie 1

Napisać w języku ANSI C program sortujący algorytmem sortowania gnomu (tzw. “gnome sort”) tablicę pól prostokątów. W tym celu należy stworzyć strukturę `Rectangle` zawierającą dwa pola typu `double` – długości boków prostokąta. Następnie należy przypisać do każdego prostokąta w tablicy odpowiednie parametry - długości boków. Później należy stworzyć tablicę typu `double` wartości pól prostokątów (które należy obliczyć korzystając z uprzednio stworzonej tablicy obiektów typu `Rectangle`).

Opis algorytmu sortowania gnomu w pseudokodzie:

```
function GnomeSort(array[0..arrayLen-1])
{
    i := 1
    j := 2
    while i < arrayLen
        if array[i-1] > array[i]
            swap array[i-1] and array[i]
            i := i - 1
            if i > 0
                continue
        i := j
        j := j + 1
}
```

Schemat działania algorytmu znajduje się na angielskiej Wikipedii:

http://en.wikipedia.org/wiki/Gnome_sort

Uwagi:

1. Do zamiany dwóch wartości liczbowych należy użyć funkcji: **1 p.**
`void swap(double *a, double *b)`
2. Do sortowania tablicy należy użyć funkcji: **1.5 p.**
`void GnomeSort(double *array, int arrayLen)`
3. Do wypisania elementów tablicy należy użyć funkcji: **1 p.**
`void Print(double *array, int arrayLen)`
4. Dynamiczne tworzenie i zwalnianie tablic, parametry (ilość prostokątów oraz długości boków) przekazywane jako parametry uruchomienia programu. **1.5 p.**
5. Program piszemy w dowolnym edytorze tekstu i kompilujemy go za pomocą kompilatora g++ (kompilator języka C++) w wierszu poleceń, np.:
`g++ -Wall program.cpp -o program`
Flaga `-Wall` włącza dodatkowe ostrzeżenia kompilatora (warnings).
6. Przykładowy output:
Pola prostokątów przed sortowaniem:
0: 9.000000
1: 4.000000
2: 36.000000
Pola prostokątów po sortowaniu:
0: 4.000000
1: 9.000000
2: 36.000000