

Algorytmy genetyczne 2017/18 - projekt

Opis problemu

Powierzono nam pracę na odpowiedzialnym stanowisku – obsługujemy piłę do cięcia płyt meblowych w markecie budowlanym. Duża płyta meblowa jest prostokątna i ma wymiary (szer. x wys.) $2800 \times 2070 \text{ mm}^2$. Mamy do wykonania zlecenie, listę mniejszych prostokątnych płyt o podanych wymiarach, które należy z tej dużej płyty wyciąć. Mamy tylko jedną dużą płytę, a mniejszych płyt może być tak dużo, że nie uda się ich wszystkich wyciąć z dużej. Musimy tak wybrać mniejsze płyty do wycięcia i rozplanować proces cięcia (tj. w którym miejscu z dużej płyty wytniemy daną małą płytę), aby zostało jak najmniej niewykorzystanego materiału, a więc odpadu. Inaczej mówiąc sumaryczne pole powierzchni S małych płyt przeznaczonych do wycięcia ma być jak największe. Piła porusza się tylko w kierunkach góra-dół i prawo-lewo, więc nie możemy żadnej małej płyty wyciąć pod kątem, ale możemy ją rozplanować w pionie lub w poziomie. Zakładamy, że szerokość piły jest pomijalnie mała

Dane wejściowe

Program wczytuje dane wejściowe z pliku `maleplyty.txt`, który w każdej linii zawiera parę liczb całkowitych a b oddzielonych spacją, oznaczających szerokość i wysokość płyty wyrażone w milimetrach. Po każdej parze liczb, włącznie z ostatnią, jest znak końca linii. Separatorem dziesiętnym jest kropka. Przykładowy plik `maleplyty.txt` jest załącznikiem do niniejszego opisu.

Dane wyjściowe

Program powinien zapisać rozwiązanie w pliku `output.txt`. W pierwszej linii powinno być zapisane sumaryczne pole powierzchni płyt S , których cięcie udało się rozplanować na dużej płycie w mm^2 . W kolejnych liniach oddzielone spacją czwórki liczb a b x y r gdzie:

- a , b – szerokość i wysokość płyty; kolejność linii musi być taka sama, jak w pliku `maleplyty.txt`
- x , y – to współrzędne lewego górnego rogu małej płyty, jeśli dana płyta będzie wycięta, lub para liczb -1 -1, jeśli danej płyty nie tnijemy. Lewemu górnemu rogowi dużej płyty odpowiadają współrzędne (0, 0)
- r – ma wartość '0', jeśli płyty nie obracamy do cięcia oraz '1', gdy obracamy ją w tym celu o 90°

Separatorem dziesiętnym jest kropka. Po każdej parze liczb, włącznie z ostatnią, jest znak końca linii. Przykładowy plik `output.txt` jest załącznikiem do niniejszego opisu.

Założenia wstępne

Liczba małych płyt N będzie nie większa niż 40. Pojedyncze wykonanie programu nie powinno trwać dłużej niż 5 minut na przeciętnym komputerze.

Projekt

Każda osoba pisze własny program, który może wykorzystywać fragmenty programów z laboratoriów. Można także napisać program od zera. Program może wykorzystywać bibliotekę GALib, ale nie musi, ocena nie będzie od tego zależała. Archiwum z projektem powinno się nazywać `AG1718_ImięNazwisko_projekt.zip`. W środku proszę umieścić:

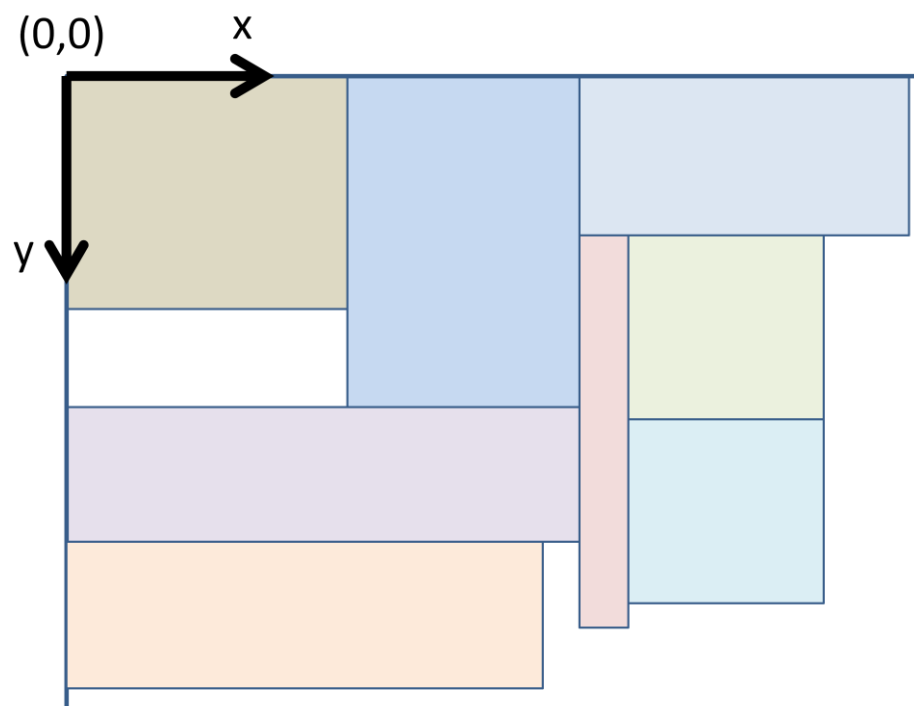
- podkatalog "src" z plikami źródłowymi oraz wszystkimi biblioteki niestandardowymi (poza GALib), które są potrzebne do skompilowania projektu
- podkatalog "exe", zawierający skompilowaną wersję programu. Ta wersja powinna działać na Windows 8.1, Windows 10 lub Ubuntu 16.04 bez konieczności instalowania żadnych dodatkowych pakietów, programów itp.
- sprawozdanie z realizacji projektu (maksimum 5 str. A4) z opisem zastosowanego kodowania, metody selekcji, funkcji dostosowania, testów, którym poddano aplikację i innych elementów, które wydają się Państwu istotne.

Ocenianie

1. Za cały projekt będzie można uzyskać maksymalnie **30 pkt.**
2. Będę oceniał poprawność implementacji, m.in. czy algorytm genetyczny działa jak należy, czy małe płyty nie zachodzą na siebie, czy nie wystają poza dużą płytę, czy dobrze liczona jest wartość S itd. Za poprawnie działający algorytm, napisany bez błędów, będzie można uzyskać maksymalnie **15 pkt.**
3. Za sprawozdanie można będzie uzyskać **6 pkt.**
4. W ocenianiu pojawi się również czynnik rywalizacji. Każdy program zostanie uruchomiony dziesięciokrotnie z różnymi plikami `maleplyty.txt` (dla wszystkich programów jednakowymi). Dla każdego uruchomienia zanotuję wyznaczoną przez program wartość S , a następnie obliczę średnią z tych 10 uruchomień S_{sr} . Następnie zbiorę wartości S_{sr} dla wszystkich programów. Osoby, które znajdą się na lub powyżej 90. centyla uzyskają **9 pkt.** Programy trafiające w lub powyżej 70. centyla uzyskają **6 pkt.** Wynik na poziomie przynajmniej 50. centyla oznacza **4 pkt.**, na 30. centylu lub wyżej **2 pkt.**, a pozostałe algorytmy nie uzyskają punktów za ten element oceny.
5. Programy z istotnymi błędami implementacji otrzymają w sumie od **0** do **10 pkt.**, niezależnie od generowanych wyników. Autorzy takich programów mogą jeszcze liczyć na punkty za sprawozdanie.

Termin oddania projektów

Gotowe projekty proszę przesłać do mnie drogą mailową w terminie do **08.01.2018 r.** do godz. **23:59**. W temacie maila proszę napisać `AG_projekt_ImięNazwisko`. Każdy dzień opóźnienia będzie skutkował obniżeniem punktacji o **4 pkt.** Prace przesłane **po 12.01.2018 r.** otrzymają **0 pkt.**



Rys. 1: Przykładowe rozplanowanie 8 małych płyt do wycięcia z dużej płyty.