

Perancangan dan Analisis Algoritma 2 (Kelas B)

Sesi Lab, 14 November 2014

NAMA: Fais Zharfan Azif

NRP: 5112100098

GREEDY-ACTIVITY-SELECTOR(s, f)

```
1   $n = s.length$ 
2   $A = \{a_1\}$ 
3   $k = 1$ 
4  for  $m = 2$  to  $n$ 
5      if  $s[m] \geq f[k]$ 
6           $A = A \cup \{a_m\}$ 
7           $k = m$ 
8  return  $A$ 
```

Di mana:

- A adalah array yang digunakan untuk menyimpan aktivitas yang dipilih.
- a_m adalah indeks aktivitas yang dipilih.
- $A \cup \{a_m\}$ adalah operasi untuk menambahkan a_m ke dalam array A .

Implementasikan algoritma *activity selector* di atas ke dalam program C/C++, dengan menggunakan *greedy choice* sebagai berikut: (program dapat dibuat dalam satu file ataupun dibuat terpisah)

- “Shortest duration” activity
- “Start first” activity
- “Finish first” activity

Petunjuk:

- Urutkan aktivitas berdasarkan *greedy choice* yang dipakai.
- Contoh input dan penentuan aktivitas yang dipilih dapat dilihat pada file Excel yang dilampirkan (Activity Selector.xlsx).

Input

- Baris pertama merupakan kode algoritma yang akan dipakai.
- Baris kedua merupakan n , yaitu jumlah aktivitas yang ada.
- n baris berikutnya, adalah pasangan waktu mulai (s_i) dan waktu berakhir (f_i) aktivitas ke- i .

Output

- baris pertama adalah durasi waktu total aktivitas yang terpilih (waktu finish aktivitas terakhir dikurangi waktu start aktivitas pertama)
- baris kedua adalah indeks aktivitas-aktivitas yang terpilih.

Input
11
1 4
3 5
0 6
5 7
3 9
4 6
6 10
8 11
8 12
2 14
12 16
Output
(jika menggunakan “shortest duration”)
13
2 4 8 11
(jika menggunakan “start first”)
16
3 7 11
(jika menggunakan “finish first”)
15
1, 6, 7, 11

Contoh dalam perhitungannya

2. Buatlah *sample case* sebanyak 3 buah dengan jumlah aktivitas minimal 10 dan cetak hasilnya pada tabel berikut.

Input
12
4 5
3 4
9 10
5 9
1 2
12 15
13 14
8 9
7 10
4 9
11 12
15 17
Output
(jika menggunakan “shortest duration”)
13
1 3 7 12
(jika menggunakan “start first”)
16
5 2 1 4 3 11 6 12
(jika menggunakan “finish first”)
16
5 2 1 4 3 11 7 12

Input
19
2 3
5 9

3 8

2 4

5 6

7 10

12 14

2 12

3 14

5 16

24 25

10 11

13 18

21 25

23 29

24 28

35 36

35 39

40 41

Output

(jika menggunakan “shortest duration”)

39

1 5 11 17 19

(jika menggunakan “start first”)

39

1 3 12 7 14 17 19

(jika menggunakan “finish first”)

39

1 5 6 12 7 11 17 19

Input

16

1 2

4 5

2 10

5 6

6 7

7 8

9 10

10 20

11 14

12 15

1 9

3 5

8 9

11 12

12 13

13 25

Output

(jika menggunakan “shortest duration”)

24

1 2 4 5 6 7 14 15 16

(jika menggunakan “start first”)

19

1 3 8

(jika menggunakan “finish first”)

24

1 2 4 5 6 13 7 14 15 16