ESc101N: Fundamentals of computing(Lab Session 7)

September 16, 2009

Instructions

1. Please read the question carefully and write the program accordingly

2. Make sure that the TA has graded you program

3. The marks are distributed as follows. You get 60% of the marks if the basic algorithm is current, 20% if you manage to compile and execute and 20% for writing the code cleanly, i.e. using proper variable names, intending and making the code more readable.

Question 1. (10 marks) Given a sequence of positive integers $\{a_1, \ldots, a_n\}$, a *bar graph* is a sequence of n columns such that the i column has exactly a_i '#''s and the rest spaces.

Write a program that will read n numbers from the user and prints its vertical bar graph.

Hint: Print stuff line by line. The `j`-th th character of the `i`-th line is either a '#' or a space character. Question is when is it '#' and when is it a space.

```
$ ./a.out
enter the sequence length: 20
enter a[0]: 1
enter a[1]: 0
enter a[2]: 3
enter a[3]: 5
enter a[4]: 9
enter a[5]: 2
enter a[6]: 4
enter a[7]: 2
enter a[8]: 9
enter a[9]: 2
enter a[10]: 5
enter a[11]: 2
enter a[12]: 5
enter a[13]: 2
enter a[14]: 1
enter a[15]: 3
enter a[16]: 5
enter a[17]: 6
enter a[18]: 4
enter a[19]: 1
the bar chart is
```

```
     #     #
     #     #
     #     #
     #     #          #
   ##    # # #    ##
    ## # # # #    ###
  ### # # # #   ####
  ########### ####
# ################
-------------------
$
```

> **Solution:**
>
> ```c
> #include <stdio.h>
> #include <stdlib.h>
> #include<unistd.h>
> #define N 100
> void printNChar(int n, char c);
> void barchart(int *, int, int, int);
>
> int main()
> {
>   int a[N];
>   int n;
>   int maxP=0, minN = 0;
>   printf("enter the sequence length: ");
>   scanf("%d", &n);
>
>   for(int i = 0 ; i < n ; i ++)
>   {
>
>     printf("enter a[%d]: ", i );
>     scanf("%d", a + i );
>     if( a[i] < 0 && a[i] < minN) {minN = a[i];}
>     if( a[i] > 0 && a[i] > maxP) {maxP = a[i];}
>   }
>
>     printf("the bar chart is \n\n");
>   barchart(a,n,maxP,minN);
>   return 0;
>
>   }
>
> void barchart(int *a, int n, int maxP, int minN)
> {
>     for(int i = maxP; i > 0 ; i--)
>     {
>
>       for(int j = 0; j < n; j++)
>       {
> ```

```
        if( a[j] >= i ) printf("#");
        else printf(" ");
      }
      printf("\n");
    }

    printNChar(n, '-');
    printf("\n");

    for(int i = -1; i >= minN ; i--)
    {

      for(int j = 0; j < n; j++)
      {
        if( i >= a[j] ) printf("#");
        else printf(" ");
      }
      printf("\n");

    }

  }
  void printNChar(int n, char c)
  {
    for(int i = 0; i < n; i++){printf("%c",c);}
    return;
  }
```

Question 2. (0 marks) Code up your favourite sorting algorithm. Use the subroutine `barchart` that you wrote in the previous assignment to display the contents of the array as a bar chart. You can perform an animation of it using the `system("clear");` command. You can wait for `n` second (so that you actually see the algorithm in action) using the command `sleep(n);`. eg

Example

```
#include <stdlib.h>
#include <unistd.h>

int main()
{
  sleep(5);
  system("clear");

}
```

will pause for 5 seconds and then clear the screen.