```c
/*
 * functions.c
 *
 *  Created on: Oct. 2, 2018
 *      Author: JJ
 *      Here I keep all the function for house keeping purposes
 */

#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <time.h>
#include <unistd.h>
#include <conio.h>

#include "functions.h"


// defining length of the beep and time pause as global
//setting their default values
float beepfreq=200.0;
float pauseLength=0.4;



/*-----------------------------------------------
--------Structural designs and handles---------
-----------------------------------------------*/



//lunches the main menu choice
void lunch(int menuId,int *key)
{

    if (menuId==101)
    {
        system("cls");
        sub1Menu();


    }
    else if (menuId==102)
    {
        system("cls");
        sub2Menu();




    }   else if (menuId==103)
```

```
54      {
55          system("cls");
56          sub3Menu();
57
58
59
60
61
62      }   else if (menuId==104)
63      {
64          system("cls");
65          sub4Menu();
66
67
68
69      }
70      else if (menuId==100)
71      {
72          system("cls");
73          *key=1000;
74
75
76      }
77
78  }
79
80  void displayMainMenu(int menuId)
81  {
82      HANDLE  hConsole;// Adjusting the envirnment color
83      hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
84      SetConsoleTextAttribute(hConsole, 15);
85      system("cls");
86      puts("");
87      puts("================================================================");
88      puts("==============  Advanced Operating System    ==================");
89      puts("============== Presented to: Dr. Ashoke Deb  ==================");
90      puts("==============          Assignment #1         ==================");
91      puts("==============  By: Seyed Javad Khataei Pour ==================");
92      puts("================================================================");
93      puts("");
94
95
96
97      if (menuId==101)
98      {
99          SetConsoleTextAttribute(hConsole, 10);
100         puts("  Subproblem one - Random Generator\n");
101         SetConsoleTextAttribute(hConsole, 15);
102         puts("  Subproblem two - Print a letter\n");
103         puts("  Subproblem three - combination of subproblems one and two
    \n");
104         puts("  Option\n");
105         puts("  Exit\n");
```

```
106
107     }
108     else if (menuId==102)
109     {
110         SetConsoleTextAttribute(hConsole, 15);
111         puts("  Subproblem one - Random Generator\n");
112         SetConsoleTextAttribute(hConsole, 10);
113         puts("  Subproblem two - Print a letter\n");
114         SetConsoleTextAttribute(hConsole, 15);
115         puts("  Subproblem three - combination of subproblems one and two
    \n");
116         puts("  Option\n");
117         puts("  Exit\n");
118
119     }   else if (menuId==103)
120     {
121         SetConsoleTextAttribute(hConsole, 15);
122         puts("  Subproblem one - Random Generator\n");
123         puts("  Subproblem two - Print a letter\n");
124         SetConsoleTextAttribute(hConsole, 10);
125         puts("  Subproblem three - combination of subproblems one and two
    \n");
126         SetConsoleTextAttribute(hConsole, 15);
127         puts("  Option\n");
128         puts("  Exit\n");
129
130     }   else if (menuId==104)
131     {
132         SetConsoleTextAttribute(hConsole, 15);
133         puts("  Subproblem one - Random Generator\n");
134         puts("  Subproblem two - Print a letter\n");
135         puts("  Subproblem three - combination of subproblems one and two
    \n");
136         SetConsoleTextAttribute(hConsole, 10);
137         puts("  Option\n");
138         SetConsoleTextAttribute(hConsole, 15);
139         puts("  Exit\n");
140     }
141     else if (menuId==100 || menuId==105)
142     {
143         SetConsoleTextAttribute(hConsole, 15);
144         puts("  Subproblem one - Random Generator\n");
145         puts("  Subproblem two - Print a letter\n");
146         puts("  Subproblem three - combination of subproblems one and two
    \n");
147         puts("  Option\n");
148         SetConsoleTextAttribute(hConsole, 10);
149         puts("  Exit\n");
150         SetConsoleTextAttribute(hConsole, 15);
151     }
152
153 }
154
```

```
155 //=====================================================
156 //handles sub menu #1. Watch for the keys user presses and
157 void sub1Menu()
158 {
159     int key1,sub1ChoiceCounter;
160 
161 
162             // this is the basic choice for main menu which is set to
    first
163             //101 is the base so when user presses the Up button and
    mainChoiceCounter decrease
164             // it will be a positive number which is easier to handle here
165             sub1ChoiceCounter=101;
166             displaysub1Menu(sub1ChoiceCounter);// display the main menu
    based on the user's choice
167 
168 
169             while ( key1 != 27)
170             {
171             key1= getch();// reading the user's key pressed
172                 if (key1==80)// The Down arrow key is pressed
173                 {
174                     sub1ChoiceCounter++;
175                     sub1ChoiceCounter=sub1ChoiceCounter%4+100;// keep the
    sub1ChoiceCounter in a certain range so we can assign different function
    for each choice easily by using if in the displaymainmenu() function
176                     displaysub1Menu(sub1ChoiceCounter);
177 
178                 }
179                  else if(key1==72)// The Up arrow key is pressed
180                 {
181 
182                         sub1ChoiceCounter--;
183                         sub1ChoiceCounter=(sub1ChoiceCounter%4+100);
184                         displaysub1Menu(sub1ChoiceCounter);
185 
186                 }
187                  else if(key1==13)// The Enter arrow key is pressed.
188                         {
189                          // Invoke different function based on the menu
    that is selected by the user
190                          lunchsub1(sub1ChoiceCounter,&key1);// Sending by
    reference so we can edit the key pressed in case of need( eg. exit)
191                          displaysub1Menu(sub1ChoiceCounter);
192 
193 
194                         }
195     }
196 }
197 
198 void displaysub1Menu(int menuId)
199 {
200     HANDLE  hConsole;
```

```
201     hConsole = GetStdHandle(STD_OUTPUT_HANDLE);// to change the text
    color. creating an illusion of selection for the user
202     system("cls");
203     puts("");
204     printf("    ============Please choose from bellow options============
    \n");
205     printf("    ================== To exit press ESC ====================
    \n");
206     puts("");
207     puts("");
208
209
210
211
212     // Based on the user choice one of the options is green
213     if (menuId==101)// menuId is the user choice which is equal to
    sub1ChoiceCounter in sub1menu() function
214     {
215         SetConsoleTextAttribute(hConsole, 10);
216         puts("  Infinite random numbers with a seed from system clock,
    within a certain range\n");// sub11()
217         SetConsoleTextAttribute(hConsole, 15);
218         puts("  Infinite random numbers with a given seed from the user,
    within a certain range\n");//sub12()
219         puts("  Finite random numbers with a seed from system clock,
    within a certain range\n"); //sub13()
220         puts("  Finite random numbers with a given seed from the user,
    within a certain range\n"); //sub14()
221
222     }
223     else if (menuId==102)
224     {
225         SetConsoleTextAttribute(hConsole, 15);
226         puts("  Infinite random numbers with a seed from system clock,
    within a certain range\n");// sub11()
227         SetConsoleTextAttribute(hConsole, 10);
228         puts("  Infinite random numbers with a given seed from the user,
    within a certain range\n");//sub12()
229         SetConsoleTextAttribute(hConsole, 15);
230         puts("  Finite random numbers with a seed from system clock,
    within a certain range\n"); //sub13()
231         puts("  Finite random numbers with a given seed from the user,
    within a certain range\n"); //sub14()
232
233     }   else if (menuId==103)
234     {
235         SetConsoleTextAttribute(hConsole, 15);
236         puts("  Infinite random numbers with a seed from system clock,
    within a certain range\n");// sub11()
237         puts("  Infinite random numbers with a given seed from the user,
    within a certain range\n");//sub12()
238         SetConsoleTextAttribute(hConsole, 10);
239         puts("  Finite random numbers with a seed from system clock,
```

```
   within a certain range\n"); //sub13()
240         SetConsoleTextAttribute(hConsole, 15);
241         puts("  Finite random numbers with a given seed from the user,
   within a certain range\n"); //sub14()
242 
243     }   else if (menuId==100)
244     {
245         SetConsoleTextAttribute(hConsole, 15);
246         puts("  Infinite random numbers with a seed from system clock,
   within a certain range\n");// sub11()
247         puts("  Infinite random numbers with a given seed from the user,
   within a certain range\n");//sub12()
248         puts("  Finite random numbers with a seed from system clock,
   within a certain range\n"); //sub13()
249         SetConsoleTextAttribute(hConsole, 10);
250         puts("  Finite random numbers with a given seed from the user,
   within a certain range\n"); //sub14()
251         SetConsoleTextAttribute(hConsole, 15);
252     }
253 
254 
255 }
256 
257 
258 //lunches the functions that are selected from submenu#1
259 void lunchsub1(int submenuId,int *key1)
260 {
261 
262     if (submenuId==101)
263     {
264         system("cls");
265 
266         //Do the sub11 function which is written notebook 24 june
267         sub11();
268 
269     }
270     else if (submenuId==102)
271     {
272         system("cls");
273         sub12();
274 
275 
276 
277 
278 
279     }   else if (submenuId==103)
280     {
281         system("cls");
282         sub13();
283 
284 
285 
286 
```

```
287
288     }   else if (submenuId==100)
289     {
290         system("cls");
291         sub14();
292
293
294     }
295
296
297 }
298
299 //================================================
300 //handles sub menu #2. Watch for the keys user presses and
301 void sub2Menu()
302 {
303     int key2,sub2ChoiceCounter;
304
305
306             // this is the basic choice for main menu which is set to
    first
307             //101 is the base so when user presses the Up button and
    ChoiceCounter decrease
308             // it will be a positive number which is easier to handle here
309             sub2ChoiceCounter=101;
310             displaysub2Menu(sub2ChoiceCounter);// display the main menu
    based on the user's choice
311
312
313             key2=100;
314             while ( key2 != 27)
315             {
316
317             key2= getch();// reading the user's key pressed
318                 if (key2 == 80)// The Down arrow key is pressed
319                 {
320
321                     sub2ChoiceCounter++;
322                     sub2ChoiceCounter=sub2ChoiceCounter % 3 +100;// keep
    the sub2ChoiceCounter in a certain range so we can assign different
    function for each choice easily by using if in the displaymainmenu()
    function
323
324                     displaysub2Menu(sub2ChoiceCounter);
325
326                 }
327                  else if(key2 == 72)// The Up arrow key is pressed
328                 {
329
330                         sub2ChoiceCounter= sub2ChoiceCounter-1;
331                         sub2ChoiceCounter=sub2ChoiceCounter%3+100;
332                         displaysub2Menu(sub2ChoiceCounter);
333
```

```
334
335
336                 }
337                  else if(key2==13)// The Enter arrow key is pressed.
338                         {
339                          // Invoke different function based on the menu
   that is selected by the user
340                          lunchsub2(sub2ChoiceCounter,&key2);// Sending by
   reference so we can edit the key pressed in case of need( eg. exit)
341                          displaysub2Menu(sub2ChoiceCounter);
342
343
344                         }
345     }
346 }
347
348 void displaysub2Menu(int menuId)
349 {
350     HANDLE  hConsole;
351     hConsole = GetStdHandle(STD_OUTPUT_HANDLE);// to change the text
   color. creating an illusion of selection for the user
352     system("cls");
353     puts("");
354     printf("    ============Please choose from bellow options============
   \n");
355     printf("    ================== To exit press ESC ====================
   \n");
356     printf("    =============== Use Down key to navigate==================
   \n");
357
358     puts("");
359     puts("");
360
361
362
363
364     // Based on the user choice one of the options is green
365     if (menuId==101)// menuId is the user choice which is equal to
   sub2ChoiceCounter in sub1menu() function
366     {
367         SetConsoleTextAttribute(hConsole, 10);
368         puts("  Execute Print A function\n");
369         SetConsoleTextAttribute(hConsole, 15);
370         puts("  Execute Print B function\n");
371         puts("  Execute Print C function\n");
372
373     }
374     else if (menuId==100)
375     {
376         SetConsoleTextAttribute(hConsole, 15);
377         puts("  Execute Print A function\n");
378         SetConsoleTextAttribute(hConsole, 10);
379         puts("  Execute Print B function\n");
```

```
380         SetConsoleTextAttribute(hConsole, 15);
381         puts("  Execute Print C function\n");
382 
383     }
384     else if (menuId==102)
385     {
386         SetConsoleTextAttribute(hConsole, 15);
387         puts("  Execute Print A function\n");
388         puts("  Execute Print B function\n");
389         SetConsoleTextAttribute(hConsole, 10);
390         puts("  Execute Print C function\n");
391         SetConsoleTextAttribute(hConsole, 15);
392     }
393 
394 
395 }
396 
397 
398 //lunches the functions that are selected from submenu#2
399 void lunchsub2(int submenuId,int *key2)
400 {
401 
402     system("cls");
403 
404     puts("");
405     puts("======To exit press any key=======");
406 
407     if (submenuId==101)
408     {
409         printA();
410 
411 
412     }
413     else if (submenuId==102)
414     {
415         printC();
416 
417 
418 
419 
420 
421     }   else if (submenuId==100)
422     {
423         printB();
424 
425 
426 
427 
428 
429     }
430     getch();// give the user time to see the result
431 
432 }
```

```
433
434
435 //================================================
436 //handles sub menu #3. Watch for the keys user presses and
437 void sub3Menu()
438 {
439     int key3,sub3ChoiceCounter;
440
441
442             // this is the basic choice for main menu which is set to
    first
443             //101 is the base so when user presses the Up button and
    mainChoiceCounter decrease
444             // it will be a positive number which is easier to handle here
445             sub3ChoiceCounter=101;
446             displaysub3Menu(sub3ChoiceCounter);// display the main menu
    based on the user's choice
447
448
449             while ( key3 != 27)
450             {
451             key3= getch();// reading the user's key pressed
452                 if (key3==80)// The Down arrow key is pressed
453                 {
454                     sub3ChoiceCounter++;
455                     sub3ChoiceCounter=sub3ChoiceCounter%2+100;// keep the
    sub1ChoiceCounter in a certain range so we can assign different function
    for each choice easily by using if in the displaymainmenu() function
456                     displaysub3Menu(sub3ChoiceCounter);
457
458                 }
459                  else if(key3==72)// The Up arrow key is pressed
460                 {
461
462                         sub3ChoiceCounter--;
463                         sub3ChoiceCounter=(sub3ChoiceCounter%2+100);
464                         displaysub3Menu(sub3ChoiceCounter);
465
466                 }
467                  else if(key3==13)// The Enter arrow key is pressed.
468                         {
469                          // Invoke different function based on the menu
    that is selected by the user
470                          lunchsub3(sub3ChoiceCounter,&key3);// Sending by
    reference so we can edit the key pressed in case of need( eg. exit)
471                          displaysub3Menu(sub3ChoiceCounter);
472
473
474                         }
475     }
476 }
477
478 void displaysub3Menu(int menuId)
```

```
479 {
480     HANDLE  hConsole;
481     hConsole = GetStdHandle(STD_OUTPUT_HANDLE);// to change the text
    color. creating an illusion of selection for the user
482     system("cls");
483     puts("");
484     printf("    ============Please choose from bellow options============
    \n");
485     printf("    ================== To exit press ESC ====================
    \n");
486     puts("");
487     puts("");
488
489
490
491
492     // Based on the user choice one of the options is green
493     if (menuId==101)// menuId is the user choice which is equal to
    sub3ChoiceCounter in sub1menu() function
494     {
495         SetConsoleTextAttribute(hConsole, 10);
496         puts("  Random sequence of A,B ,and C with a seed obtained from
    system clock\n");// sub11()
497         SetConsoleTextAttribute(hConsole, 15);
498         puts("  Random sequence of A,B ,and C with a seed obtained from
    the user\n");//sub12()
499     }
500     else if (menuId==100)
501     {
502         SetConsoleTextAttribute(hConsole, 15);
503         puts("  Random sequence of A,B ,and C with a seed obtained from
    system clock\n");// sub11()
504         SetConsoleTextAttribute(hConsole, 10);
505         puts("  Random sequence of A,B ,and C with a seed obtained from
    the user\n");//sub12()
506         SetConsoleTextAttribute(hConsole, 15);
507     }
508
509
510 }
511
512
513 //lunches the functions that are selected from submenu#3
514 void lunchsub3(int submenuId,int *key3)
515 {
516
517     if (submenuId==101)
518     {
519         system("cls");
520         sub31();// sub31 and 32 are functions that calculate and print out
    the results
521
522     }   else if (submenuId==100)
```

```
523     {
524         system("cls");
525         sub32();
526     }
527 }
528 
529 //================================================
530 //handles sub menu #4. Watch for the keys user presses and
531 void sub4Menu()
532 {
533     int key4,sub4ChoiceCounter;
534 
535 
536             // this is the basic choice for main menu which is set to
    first
537             //101 is the base so when user presses the Up button and
    mainChoiceCounter decrease
538             // it will be a positive number which is easier to handle here
539             sub4ChoiceCounter=101;
540             displaysub4Menu(sub4ChoiceCounter);// display the main menu
    based on the user's choice
541 
542 
543             while ( key4 != 27)
544             {
545             key4= getch();// reading the user's key pressed
546                 if (key4==80)// The Down arrow key is pressed
547                 {
548                     sub4ChoiceCounter++;
549                     sub4ChoiceCounter=sub4ChoiceCounter%2+100;// keep the
    sub1ChoiceCounter in a certain range so we can assign different function
    for each choice easily by using if in the displaymainmenu() function
550                     displaysub4Menu(sub4ChoiceCounter);
551 
552                 }
553                  else if(key4==72)// The Up arrow key is pressed
554                 {
555 
556                         sub4ChoiceCounter--;
557                         sub4ChoiceCounter=(sub4ChoiceCounter%2+100);
558                         displaysub4Menu(sub4ChoiceCounter);
559 
560                 }
561                  else if(key4==13)// The Enter arrow key is pressed.
562                        {
563                         // Invoke different function based on the menu
    that is selected by the user
564                          lunchsub4(sub4ChoiceCounter,&key4);// Sending by
    reference so we can edit the key pressed in case of need( eg. exit)
565                          displaysub4Menu(sub4ChoiceCounter);
566 
567 
568                         }
```

```
569     }
570 }
571
572 void displaysub4Menu(int menuId)
573 {
574     HANDLE  hConsole;
575     hConsole = GetStdHandle(STD_OUTPUT_HANDLE);// to change the text
    color. creating an illusion of selection for the user
576     system("cls");
577     puts("");
578     printf("    ============Please choose from bellow options============
    \n");
579     printf("    ================== To exit press ESC ====================
    \n");
580     puts("");
581     puts("");
582
583
584
585
586     // Based on the user choice one of the options is green
587     if (menuId==101)// menuId is the user choice which is equal to
    sub4ChoiceCounter in sub1menu() function
588     {
589         SetConsoleTextAttribute(hConsole, 10);
590         puts("  Change beep frequency. To mute enter 0\n");// sub41()
591         SetConsoleTextAttribute(hConsole, 15);
592         puts("  Change pause duration\n");//sub42()
593     }
594     else if (menuId==100)
595     {
596         SetConsoleTextAttribute(hConsole, 15);
597         puts("  Change beep frequency. To mute enter 0\n");// sub41()
598         SetConsoleTextAttribute(hConsole, 10);
599         puts("  Change pause duration\n");//sub42()
600         SetConsoleTextAttribute(hConsole, 15);
601     }
602
603
604 }
605
606
607 //lunches the functions that are selected from submenu#4
608 void lunchsub4(int submenuId,int *key4)
609 {
610
611     if (submenuId==101)
612     {
613         system("cls");
614         sub41();// sub31 and 32 are functions that calculate and print out
    the results
615
616     }   else if (submenuId==100)
```

```
617     {
618         system("cls");
619         sub42();
620     }
621 }
622 
623 
624 
625 
626 
627 //=====================================================
628 //-------------------------------------------------------------
629 //------functional(computational) functions parts-------------
630 //-------------------------------------------------------------
631 
632 
633 /*
634 
    ================================================================================
    ==
635  Name        : Submenu 1, choice 1
636  Author      : Javad
637  Description : infinite sequence of random integers within the given range
    and
638                a seed from the system clock
639 
    ================================================================================
    =
640  */
641 
642 // will be executed when the first option of sub menu one is selected
643 
644 void sub11(void)
645 {
646 
647     int keypressed,randomNumber,range;
648     char inputstr[33];// for input of the range
649 
650     puts(" ***To stop press ESC button**** ");
651     range=0;// this loop prevent getting zero as a range which can crash
    the whole program
652     while(range<1)
653     {
654         printf("Enter an integer as the upper limit. random numbers will
    be within [1, limit]: ");
655         scanf("%s",&inputstr);// Reading the range
656         sscanf(inputstr, "%d", &range);
657     }
658 
659     keypressed=100;//  Setting an initial amount for the pressed key the
    beginning so the loop can start
660     srand(time(NULL));//setting the seed based on the clock
661     while (keypressed !=27)// Keep doing il the ESC key is pressed
```

```
662     {
663             if (kbhit())// Checks if a key is pressed?
664             {
665                 keypressed= getch();// a key is pressed. here it reads it
666             }
667             randomNumber= rand()%range;
668         printf("%d ," ,randomNumber);
669         // Producing a beep sound and a pause to keep the program user-
    friendly and pleasant to work with
670         Beep(beepfreq,100);
671         sleep(pauseLength);
672     }
673     puts("");
674     puts("Operation finished. To exit press any key");
675     getch();
676
677
678 }
679
680
681 /*
682
    ================================================================================
    ==
683  Name        : Submenu 1, choice 2
684  Author      : Javad
685  Description : infinite sequence of random integers within the given range
    and
686                seed
687
    ================================================================================
    =
688  */
689
690 // will be executed when the second option of sub menu one is selected
691
692 void sub12(void)
693 {
694
695     int keypressed,randomNumber,range,seed;
696
697     puts(" ***To stop press ESC button****  ");
698
699     range=0;// this loop prevent getting zero as a range which can crash
    the whole program
700     while(range<1)
701     {
702         //reading the range and the seed for random function
703         printf("Enter an integer as the upper limit. random numbers will
    be within [1, limit] range ");
704         scanf("%d", &range);
705     }
706     printf("Enter a seed:");
```

```
707     scanf("%d", &seed);
708
709
710
711     keypressed=100;//  Setting an initial amount for the pressed key the
    beginning so the loop can start
712     srand(seed);//setting the seed based on the clock
713     while (keypressed !=27)// Keep doing il the ESC key is pressed
714     {
715             if (kbhit())// Checks if a key is pressed?
716             {
717                 keypressed= getch();// a key is pressed. here it reads it
718             }
719             randomNumber= rand()%range;
720         printf("%d ," ,randomNumber);
721         // Producing a beep sound and a pause to keep the program user-
    friendly and pleasant to work with
722         Beep(beepfreq,99);
723         sleep(pauseLength);
724     }
725     puts("");
726     puts("Operation finished. To exit press any key");
727     getch();
728
729 }
730
731
732 /*
733
    ================================================================================
    ==
734  Name        : Submenu 1, choice 3
735  Author      : Javad
736  Description : finite sequence of random integers within the given range
    and
737                a seed from the system clock
738
    ================================================================================
    =
739  */
740
741 // will be executed when the third option of sub menu one is selected
742
743 void sub13(void)
744 {
745
746     int keypressed,randomNumber,range,quantity;
747     char inputstr[33];// for input of the range
748     int i=0;
749     printf(" ***To stop press ESC button**** \n");
750
751     range=0;// this loop prevent getting zero as a range which can crash
    the whole program
```

```
752     while(range<1)
753     {
754     //reading the range and the seed for random function
755     printf("Enter an integer as the upper limit. random numbers will be
    within [1, limit] range ");
756     scanf("%d", &range);
757     }
758     printf("Enter the quantity of the random numbers:");
759     scanf("%d", &quantity);
760
761
762     srand(time(NULL));//setting the seed based on the clock
763
764         while(keypressed !=27 && i<quantity)// Keep doing till the ESC key
    is pressed
765         {
766             i++;
767             if (kbhit())// Checks if a key is pressed?
768             {
769                 keypressed= getch();// a key is pressed. here it reads it
770             }
771             randomNumber= rand()%range;
772             printf("%d ," ,randomNumber);
773             // Producing a beep sound and a pause to keep the program
    user-friendly and pleasant to work with
774             Beep(beepfreq,99);
775             sleep(pauseLength);
776         }
777
778         puts("");
779         puts("Operation finished. To exit press any key");
780         getch();
781 }
782
783
784 /*
785
    ================================================================================
    ==
786  Name        : Submenu 1, choice 4
787  Author      : Javad
788  Description : finite sequence of random integers within the given range
    and
789                a given seed
790
    ================================================================================
    =
791  */
792
793 // will be executed when the third option of sub menu one is selected
794
795 void sub14(void)
796 {
```

```
797
798     int keypressed,randomNumber,range,quantity,seed;
799     int i=0;
800     printf(" ***To stop press ESC button**** \n");
801
802     range=0;// this loop prevent getting zero as a range which can crash
    the whole program
803     while(range<1)
804     {
805     //reading the range and the seed for random function
806     printf("Enter an integer as the upper limit. random numbers will be
    within [1, limit] range ");
807     scanf("%d", &range);
808     }
809     printf("Enter the quantity of the random numbers:");
810     scanf("%d", &quantity);
811
812     printf("Enter a seed:");
813     scanf("%d", &seed);
814
815
816     srand(seed);//setting the seed based on the clock
817
818         while(keypressed !=27 && i<quantity)// Keep doing till the ESC key
    is pressed
819         {
820             i++;
821             if (kbhit())// Checks if a key is pressed?
822             {
823                 keypressed= getch();// a key is pressed. here it reads it
824             }
825             randomNumber= rand()%range;
826             printf("%d ," ,randomNumber);
827             // Producing a beep sound and a pause to keep the program
    user-friendly and pleasant to work with
828             Beep(beepfreq,99);
829             sleep(pauseLength);
830         }
831
832         puts("");
833         puts("Operation finished. To exit press any key");
834         getch();
835 }
836
837
838
839 /* Sub Problem #2 functions*/
840 // Prints character A
841 void printA()
842 {
843     printf("A");
844 }
845
```

```
846 // Prints character A
847
848 void printB()
849 {
850     printf("B");
851 }
852
853 // Prints character A
854
855 void printC()
856 {
857     printf("C");
858 }
859
860
861
862
863
864
865 /*
866
    ================================================================================
    ==
867  Name        : Submenu 1, choice 1
868  Author      : Javad
869  Description : infinite sequence of random integers within the given range
    and
870                a seed from the system clock
871
    ================================================================================
    =
872  */
873
874 // will be executed when the first option of sub menu three is selected
875
876 void sub31(void)
877 {
878
879     int keypressed,randomNumber;
880
881
882     puts(" ***To stop press ESC button**** ");
883     keypressed=100;//  Setting an initial amount for the pressed key the
    beginning so the loop can start
884     srand(time(NULL));//setting the seed based on the clock
885     while (keypressed !=27)// Keep doing till the ESC key is pressed
886     {
887             if (kbhit())// Checks if a key is pressed?
888             {
889                 keypressed= getch();// a key is pressed. here it reads it
890             }
891             randomNumber= rand()%3;// based on a random number invoke
    either printA printB or PrintC
```

```
892             if (randomNumber==0)
893             {
894                 printA();
895             }
896             else if (randomNumber==1)
897             {
898                 printB();
899             }
900             else if (randomNumber==2)
901             {
902                 printC();
903             }
904         // Producing a beep sound and a pause to keep the program user-
    friendly and pleasant to work with
905         Beep(beepfreq,99);
906         sleep(pauseLength);
907     }
908     puts("");
909     puts("Operation finished. To exit press any key");
910     getch();
911 
912 
913 }
914 //=======================================
915 
916 /*
917 
    ================================================================================
    ==
918  Name        : Submenu 3, choice 2
919  Author      : Javad
920  Description : infinite sequence of random integers within the given range
    and
921                seed
922 
    ================================================================================
    =
923  */
924 
925 // will be executed when the second option of sub menu three is selected
926 
927 void sub32(void)
928 {
929 
930     int keypressed,randomNumber,seed;
931 
932 
933         printf("Enter a seed:");
934         scanf("%d",&seed);
935         puts("");
936         puts(" ***To stop press ESC button**** ");
937         keypressed=100;//  Setting an initial amount for the pressed key
    the beginning so the loop can start
```

```
938         srand(seed);//setting the seed based on the clock
939         while (keypressed !=27)// Keep doing till the ESC key is pressed
940         {
941                 if (kbhit())// Checks if a key is pressed?
942                 {
943                     keypressed= getch();// a key is pressed. here it reads
    it
944                 }
945                 randomNumber= rand()%3;// based on a random number invoke
    either printA printB or PrintC
946                 if (randomNumber==0)
947                 {
948                     printA();
949                 }
950                 else if (randomNumber==1)
951                 {
952                     printB();
953                 }
954                 else if (randomNumber==2)
955                 {
956                     printC();
957                 }
958             // Producing a beep sound and a pause to keep the program
    user-friendly and pleasant to work with
959             Beep(beepfreq,99);
960             sleep(pauseLength);
961         }
962         puts("");
963         puts("Operation finished. To exit press any key");
964         getch();
965
966
967 }
968
969 //====================
970
971 void sub41(void)
972 {
973     //sets the beep length
974     printf("Enter beep frequency( for example 500). To mute enter 0:");
975     scanf("%f", &beepfreq);
976 }
977
978 void sub42(void)
979 {
980     //sets the pauses between each number or letter print
981     printf("Enter pause duration in milliseconds:");
982     scanf("%f", &pauseLength);//note that sleep function in c uses time
    based on second not millisecond so it has to  be devided by 1000
983     pauseLength= pauseLength/1000;
984 }
985
```