



Erzurum Atatürk University
Faculty of Engineering
Computer Engineering Department
CEN 104: Object Oriented Programming - Spring 2021

PROGRAMMING PROJECT 1
(Due Date: April 25th, 2021 - 23:59)

In your 1st programming project, you are expected to perform the game of chance called Tombala (Turkish Bingo) as a C++ console application.

The details of this game played in real life are listed below:

- 1) Tombala is a game of chance played over 3 sets with at least 2 and at most 5 players.
- 2) In the Tombala game, each player is given a unique (different) Tombala card with 15 integers on it at the start of the game. These numbers are random numbers in the range of 1 (inclusive) to 90 (inclusive).
- 3) Each set of the game uses a bag of numbers in the range of 1 (included) and 90 (included). A number is randomly drawn from this bag each time, and each player checks if the number exists on their Tombala card. This process continues similarly until the game set is completed.
- 4) The first player with 5 numbers on his card is deemed to have made the first “Çinko”, the first player having the card with 10 numbers drawn from the bag makes the second “Çinko”, and the player with all the numbers on his card drawn from the bag is deemed to have made “Tombala”.
- 5) The player who makes the first “Çinko” gets 5 points, the player who makes the second “Çinko” gets 10 points, and the player who makes the Tombala gets 15 points.
- 6) When any player makes Tombala, the game set ends.
- 7) The player with the highest score after 3 sets of games wins the game.

To simulate the game detailed above in a computer environment as a C++ console application, follow the steps below:

- i. Declare a two-dimensional array of 10 rows and 15 columns that can store integers in the **int** data type. Each row of this two-dimensional array represents a Tombala card. Therefore, this array has the capacity to store 10 different Tombala cards.
- ii. Assign 15 unique random integers in the range of 1 (inclusive) to 90 (inclusive) to each row of this array. For this, use the Function Prototype:

void KartlariOlustur(int[][15], int);

The first parameter of this function is the two-dimensional array that stores Tombala cards, and the second parameter is the total number of cards. Then print the contents of the array on the screen similar to the following output.

TOMBALA KARTLARI HAZIRLANIYOR...															
Kart No0:	49	78	3	59	83	40	87	81	13	67	52	74	32	26	90
Kart No1:	50	82	75	20	62	21	18	66	79	48	81	85	59	1	35
Kart No2:	23	60	27	62	61	75	53	73	63	81	86	46	3	80	71
Kart No3:	3	39	24	21	85	90	22	10	47	13	56	15	65	34	27
Kart No4:	61	56	11	23	90	35	14	86	42	68	75	74	32	7	70
Kart No5:	5	53	57	75	62	13	49	79	80	23	19	67	74	39	34
Kart No6:	74	9	31	25	38	67	70	51	49	18	30	11	14	34	48
Kart No7:	24	23	37	46	63	77	26	39	88	10	31	45	22	1	75
Kart No8:	9	38	51	69	17	63	45	28	80	74	90	49	34	21	72
Kart No9:	18	19	80	42	6	15	3	76	28	85	30	50	66	20	84

Eng: Loading Tombala Cards... / Tr: Tombala Kartları Hazırlanıyor...
Eng: Card number / Tr: Kart No

- iii. Get the number of players (minimum 2 and maximum 5) from the user through the keyboard and randomly assign one of the cards stored in the two-dimensional array to each player. Then, print the players' cards on the screen similar to the following output.

```
Oyuncu Sayisi:5
Oyuncular için secilen kart numaralari
5 7 8 1 9
0. Oyuncunun Tombala Karti:  5  53  57  75  62  13  49  79  80  23  19  67  74  39  34
1. Oyuncunun Tombala Karti: 24  23  37  46  63  77  26  39  88  10  31  45  22  1  75
2. Oyuncunun Tombala Karti:  9  38  51  69  17  63  45  28  80  74  90  49  34  21  72
3. Oyuncunun Tombala Karti: 50  82  75  20  62  21  18  66  79  48  81  85  59  1  35
4. Oyuncunun Tombala Karti: 18  19  80  42  6  15  3  76  28  85  30  50  66  20  84
```

Eng: Number of Players / Tr: Oyuncu Sayısı
Eng: Card numbers selected for players / Tr: Oyuncular için seçilen kart numaraları
Eng: Player's Tombala Card / Tr: Oyuncunun Tombala Karti

- iv. Declare a 90-element one-dimensional array that stores integers in the int data type to represent the Tombala bag in a computer environment. Assign unique random integers in the range of 1 (inclusive) to 90 (inclusive) to each element of this array. Then print the contents of this array on the screen similar to the following output.

```
CEKILEN NUMARALAR:
74  17  28  26  24  47  13  70  2  33
68  51  80  77  61  37  53  72  9  88
63  81  14  38  56  60  73  49  58  32
64  82  30  15  69  75  66  31  84  27
16  46  57  36  40  48  55  34  8  79
7  85  12  45  20  43  19  83  11  29
18  22  78  3  62  21  65  41  10  4
6  90  1  89  52  5  25  35  86  23
76  44  39  54  87  71  59  42  67  50
```

Eng: Numbers Drawn / Tr: Çekilen Numaralar

- v. The first element of the one-dimensional array represents the first number drawn randomly from the bag during normal game, and likewise, the second element represents the second number to be drawn from the bag. Therefore, continue the game by taking into account all the elements of this array one by one and checking them over the Tombala cards of all players. For example, the output above tells us that the first number to search for Tombala cards is 74 and the second number is 17.
- vi. The first player with 5 numbers on his card drawn from the Tombala bag makes the first “Çinko” and gets 5 points. When the first “Çinko” is completed, print on the screen similar to the output following which player made the first “Çinko” and the final status of the players’ cards. Use the number “-1” on the cards instead of the numbers drawn from the bag.

Birinci Cinko kazanan 2 numarali oyuncu														
5	53	57	75	62	-1	49	79	-1	23	19	67	-1	39	34
-1	23	37	46	63	77	-1	39	88	10	31	45	22	1	75
9	38	-1	69	-1	63	45	-1	-1	-1	90	49	34	21	72
50	82	75	20	62	21	18	66	79	48	81	85	59	1	35
18	19	-1	42	6	15	3	76	-1	85	30	50	66	20	84

Eng: Winner of the first “Çinko” is Player 2 / Tr: Birinci Cinko kazanan 2 numaralı oyuncu

- vii. The first player with 10 numbers on his card drawn from the Tombala bag makes the second “Çinko” and gets 10 points. When the second “Çinko” is completed, print on the screen similar to the output following which player made the second “Çinko” and the final status of the players’ cards. Use the number “-1” on the cards in place the numbers drawn from the bag.

İkinci Cinko kazanan 2 numarali oyuncu														
5	-1	57	75	62	-1	-1	79	-1	23	19	67	-1	39	34
-1	23	-1	46	-1	-1	-1	39	-1	10	31	45	22	1	75
-1	-1	-1	69	-1	-1	45	-1	-1	-1	90	-1	34	21	-1
50	82	75	20	62	21	18	66	79	48	-1	85	59	1	35
18	19	-1	42	6	15	3	76	-1	85	30	50	66	20	84

Eng: Winner of the second “Çinko” is Player 2 / Tr: İkinci Cinko kazanan 2 numaralı oyuncu

- viii. The first player with 15 (All) numbers on his card drawn from the Tombala bag makes the Tombala and gets 15 points. When the Tombala is made, print on the screen similar to the output following which player made the Tombala and the final status of the players’ cards. Use the number “-1” on the cards in place the numbers drawn from the bag.

Tombala Yapan 2 numarali oyuncu														
5	-1	-1	-1	-1	-1	-1	-1	-1	23	-1	67	-1	39	-1
-1	23	-1	-1	-1	-1	-1	39	-1	-1	-1	-1	-1	1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
50	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	59	1	35
-1	-1	-1	42	-1	-1	-1	76	-1	-1	-1	50	-1	-1	-1

Eng: Winner of the Tombala is Player 2 / Tr: Tombala yapan 2 numaralı oyuncu

- ix. Similar to the above, 2 more sets (3 sets in total) must be played. In each new set, players must be assigned different random Tombala cards and the Tombala bag must be filled with different random numbers.

- x. At the end of the game, print the total points of the players and the winning player (the player with the highest score) on the screen.

RULES and GUIDELINES

1) Uygulamanızı **bir geliştirme ortamı (IDE)** kullanarak yazmanız beklenmektedir.

2) Your program should be in the following order:

Description about the application

Library Declaration

Function Prototypes

main Function

Function Definitions

3) **Description about the application** shall include

- your name,
- your lastname,
- your student number,
- your department,
- name and code of the course,
- academic semester,
- project number and
- name of the Lecturer shall be **given in comment lines.**

4) Your application must contain **at least 4** sub-functions. **At least 2** of these functions should take two-dimensional or one-dimensional arrays as parameters.

5) It is **not allowed** to use global variables. (Variables defined before **main** are called global variables. In other words, variables whose scope are both **main** functions and other functions are called global variables.)

6) C style library declaration is **not allowed**. All library declarations must be in C++ style. For example, **<ctime>** should be used instead of **<time.h>**.

7) Relevant libraries must be included in your application. For example, in order to use the **rand** and **srand** functions, the **<cstdlib>** library must be included to your program.

8) Camel notation should be used in variable naming and pascal notation should be used in function naming.

9) Any use of recursive functions is **not allowed**.

10) Changing the program flow using the **goto** statement is **not allowed**.

11) Comment lines **must** be added to the program where necessary.

12) The writing style of the development environment should not be interfered with. Tabs should be used properly.

13) You must name the application you have prepared as **FirstnameLastname_StudentNumber_CEN104_Project1.cpp** and upload it to **DBS** (Ders Bilgi

Sistemi) until the deadline. (Submission of any file or compressed file other than the .cpp file will **not be accepted.**)

14) The project you will submit should **ONLY** be the product of your own efforts. Do not forget that very serious sanctions await those who act on the contrary.

15) Do not forget that all codes submitted by students in **A, B, C and English teaching groups** will be tested with a special software for similarities.

16) Please note that projects that do not comply with the project delivery rules and/or contain plagiarism will not be evaluated.

*Turkish description is provided below.

1. programlama projenizde **Tombala** isimli şans oyununu bir C++ konsol uygulaması şeklinde gerçekleştirmeniz beklenmektedir.

Gerçek hayatta oynanan bu oyunun detayları aşağıda maddeler halinde listelenmiştir:

- 1) Tombala, en az 2 ve en fazla 5 oyuncu ile 3 set üzerinden oynanan bir şans oyunudur.
- 2) Tombala oyununda her bir oyuncuya oyun başlangıcında üzerinde benzersiz (birbirinden farklı) 15 tam sayı yazılı bir tombala kartı verilir. Söz konusu bu tam sayılar 1 (dahil)-90(dahil) aralığında rastgele sayılardır.
- 3) Oyunun her bir setinde içerisinde 1 (dahil)-90(dahil) aralığında ardışık tam sayılar yazılı pullar bulunan bir torbadan faydalanılır. Bu torbadan her seferinde rastgele bir pul çekilir ve her bir oyuncu söz konusu pul üzerindeki rakamın kendi tombala kartında var olup olmadığını kontrol eder. Bu işlem oyun seti tamamlanıncaya kadar benzer şekilde devam eder.
- 4) Kartındaki 5 tane sayı torbadan çekilen oyuncu birinci çinkoyu, 10 tane sayı torbadan çekilen oyuncu ikinci çinkoyu ve kartındaki tüm sayılar torbadan çekilmiş oyuncu da tombala yapmış kabul edilir.
- 5) Birinci çinkoyu yapan oyuncu 5 puan, ikinci çinkoyu yapan oyuncu 10 puan ve tombala yapan oyuncu da 15 puan kazanır.
- 6) Herhangi bir oyuncu tombala yaptığında ilgili oyun seti sonlanır.
- 7) 3 set oyun sonrası en yüksek puanı toplayan oyuncu oyunu kazanır.

Yukarıda detayları verilen oyunu bir C++ konsol uygulaması şeklinde bilgisayar ortamında simüle etmek için aşağıdaki adımları sırasıyla takip ediniz:

- i. 10 satırlı ve 15 sütunlu, içerisinde **int** veri tipinde tam sayılar saklayabilen iki boyutlu bir dizi deklare ediniz. İki boyutlu bu dizinin her bir satırı bir tombala kartını temsil etmektedir. Dolayısıyla bu dizi 10 farklı tombala kartı saklayabilecek kapasitedir.
- ii. Bu dizinin her bir satırına 1 (dahil)-90(dahil) aralığında 15 benzersiz rastgele tam sayı atayınız. Bu amaçla prototibi

void KartlariOlustur(int[][15], int);

şeklinde verilen alt fonksiyondan faydalanınız. Bu fonksiyonun ilk parametresi tombala kartlarını saklayan iki boyutlu diziyi ve ikinci parametre de toplam kart sayısını ifade etmektedir. Sonrasında dizi içeriğini takip eden çıktıya benzer şekilde ekrana yazdırınız.

TOMBALA KARTLARI HAZIRLANIYOR...															
Kart No0:	49	78	3	59	83	40	87	81	13	67	52	74	32	26	90
Kart No1:	50	82	75	20	62	21	18	66	79	48	81	85	59	1	35
Kart No2:	23	60	27	62	61	75	53	73	63	81	86	46	3	80	71
Kart No3:	3	39	24	21	85	90	22	10	47	13	56	15	65	34	27
Kart No4:	61	56	11	23	90	35	14	86	42	68	75	74	32	7	70
Kart No5:	5	53	57	75	62	13	49	79	80	23	19	67	74	39	34
Kart No6:	74	9	31	25	38	67	70	51	49	18	30	11	14	34	48
Kart No7:	24	23	37	46	63	77	26	39	88	10	31	45	22	1	75
Kart No8:	9	38	51	69	17	63	45	28	80	74	90	49	34	21	72
Kart No9:	18	19	80	42	6	15	3	76	28	85	30	50	66	20	84

- iii. Kullanıcıdan klavye aracılığıyla oyuncu sayısını (minimum 2 ve maksimum 5) alınız ve her bir oyuncuya iki boyutlu dizi içerisinde saklanan kartlardan birisini rastgele atayınız. Sonrasında ilgili kartları takip eden çıktıya benzer şekilde ekrana yazdırınız.

Oyuncu Sayisi:5

Oyuncular için secilen kart numaraları

5 7 8 1 9

0. Oyuncunun Tombala Kartı:	5	53	57	75	62	13	49	79	80	23	19	67	74	39	34
1. Oyuncunun Tombala Kartı:	24	23	37	46	63	77	26	39	88	10	31	45	22	1	75
2. Oyuncunun Tombala Kartı:	9	38	51	69	17	63	45	28	80	74	90	49	34	21	72
3. Oyuncunun Tombala Kartı:	50	82	75	20	62	21	18	66	79	48	81	85	59	1	35
4. Oyuncunun Tombala Kartı:	18	19	80	42	6	15	3	76	28	85	30	50	66	20	84

- xi. Tombala torbasını bilgisayar ortamında temsil etmek üzere içerisinde **int** veri tipinde tam sayılar saklayan 90 elemanlı tek boyutlu bir dizi deklare ediniz. Bu dizinin her bir elemanına 1 (dahil)-90(dahil) aralığında ve benzersiz rastgele tam sayılar atayınız. Sonrasında bu dizinin içeriğini takip eden çıktıya benzer şekilde ekrana yazdırınız.

ÇEKİLEN NUMARALAR:

74	17	28	26	24	47	13	70	2	33
68	51	80	77	61	37	53	72	9	88
63	81	14	38	56	60	73	49	58	32
64	82	30	15	69	75	66	31	84	27
16	46	57	36	40	48	55	34	8	79
7	85	12	45	20	43	19	83	11	29
18	22	78	3	62	21	65	41	10	4
6	90	1	89	52	5	25	35	86	23
76	44	39	54	87	71	59	42	67	50

- xii. Tek boyutlu dizinin ilk elemanı normal oyun esnasında torbadan rastgele çekilen ilk sayıyı, benzer şekilde ikinci elemanı torbadan çekilecek ikinci sayıyı ifade etmektedir. Dolayısıyla bu dizinin tüm elemanlarını baştan sona tek tek dikkate alarak ve tüm oyuncuların tombala kartları üzerinde gezdirerek oyuna devam ediniz. Örneğin yukarıdaki ekran çıktısı bize tombala kartlarında aranacak ilk sayının 74 ve ikinci sayının da 17 olduğunu söylemektedir.
- xiii. Kartındaki 5 sayı tombala torbasından çekilen ilk oyuncu birinci çinkoyu yapar ve 5 puan kazanır. Birinci çinko tamamlandığında hangi oyuncunun birinci çinko yaptığını ve oyuncuların kartlarının son durumunu takip eden çıktıya benzer şekilde ekrana yazdırınız. Torbadan çekilen sayıların yerine kartlar üzerinde -1 rakamını kullanınız.

Birinci Çinko kazanan 2 numaralı oyuncu

5	53	57	75	62	-1	49	79	-1	23	19	67	-1	39	34
-1	23	37	46	63	77	-1	39	88	10	31	45	22	1	75
9	38	-1	69	-1	63	45	-1	-1	-1	90	49	34	21	72
50	82	75	20	62	21	18	66	79	48	81	85	59	1	35
18	19	-1	42	6	15	3	76	-1	85	30	50	66	20	84

- xiv. Kartındaki 10 sayı tombala torbasından çekilen ilk oyuncu ikinci çinkoyu yapar ve 10 puan kazanır. İkinci çinko tamamlandığında hangi oyuncunun ikinci çinko yaptığını ve oyuncuların kartlarının son durumunu takip eden çıktıya benzer şekilde ekrana yazdırınız. Torbadan çekilen sayıların yerine kartlar üzerinde -1 rakamını kullanınız.

İkinci Cinko kazanan 2 numaralı oyuncu

5	-1	57	75	62	-1	-1	79	-1	23	19	67	-1	39	34
-1	23	-1	46	-1	-1	-1	39	-1	10	31	45	22	1	75
-1	-1	-1	69	-1	-1	45	-1	-1	-1	90	-1	34	21	-1
50	82	75	20	62	21	18	66	79	48	-1	85	59	1	35
18	19	-1	42	6	15	3	76	-1	85	30	50	66	20	84

- xv. Kartındaki 15 sayı tombala torbasından çekilen ilk oyuncu tombala yapar ve 15 puan kazanır. Bu şekilde oyunun bir seti tamamlanmış olur. Tombala sonrası hangi oyuncunun tombala yaptığını ve oyuncuların kartlarının son durumunu takip eden çıktıya benzer şekilde ekrana yazdırınız. Torbadan çekilen sayıların yerine kartlar üzerinde -1 rakamını kullanınız.

Tombala Yapan 2 numaralı oyuncu

5	-1	-1	-1	-1	-1	-1	-1	-1	23	-1	67	-1	39	-1
-1	23	-1	-1	-1	-1	-1	39	-1	-1	-1	-1	-1	1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
50	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	59	1	35
-1	-1	-1	42	-1	-1	-1	76	-1	-1	-1	50	-1	-1	-1

- xvi. Yukarıda anlatılanlara benzer şekilde 2 set daha, yani toplamda 3 set oyun oynanmalıdır. Her bir yeni sette oyunculara farklı rastgele tombala kartları atanmalıdır ve tombala torbası farklı rastgele tam sayılarla doldurulmalıdır.
- xvii. Oyun sonunda oyuncuların toplam puanlarını ve kazanan oyuncunun (en yüksek puana sahip oyuncunun) numarasını ekrana yazdırınız.

GENEL HUSUSLAR

1) Uygulamanızı **bir geliştirme ortamı (IDE)** kullanarak yazmanız beklenmektedir.

2) Programınızın

Uygulama ile İlgili Genel Açıklamalar

Kütüphane Deklarasyonu

Fonksiyon Prototipleri

main Fonksiyonu

Fonksiyon Tanımları

sırasıyla oluşturulması gerekmektedir.

3) **Uygulama ile İlgili Genel Açıklamalar** bölümünde (dosyanın en üst bölümünde)

- adınız,
- soyadınız,
- öğrenci numaranız,
- bölümünüz,
- dersin adı ve kodu,
- ayrıca dersin verildiği akademik dönem,
- projenin numarası ve
- dersi veren öğretim üyesinin adı **açıklama satırı şeklinde ver almalıdır.**

4) Uygulamanızın **en az 4** adet alt fonksiyon içermesi gerekmektedir. Bu fonksiyonlardan **en az 2** tanesi iki boyutlu ya da tek boyutlu dizileri kendisine parametre olarak almalıdır.

- 5) Global deęiřken kullanımına izin **yoktur.** (**main** öncesinde tanımlanan deęiřkenlere global deęiřken denir. Başka bir deyiřle ömrü/faaliyet alanı/sıkobu hem **main** fonksiyonu hem de dięer fonksiyonlar olan deęiřkenler global deęiřken olarak adlandırılırlar.)
- 6) C sitil kütüphane deklarasyonuna izin **yoktur.** Bütün kütüphane deklarasyonları C++ sitilinde olmalıdır. Örneęin **<time.h>** yerine **<ctime>** kullanılmalıdır.
- 7) Uygulamanızda ilgili kütüphaneler muhakkak **include** edilmelidir. Örneęin **rand** ve **srand** fonksiyonlarını kullanabilmek için programınıza **<cstdlib>** kütüphanesi eklenmelidir.
- 8) Deęiřken isimlendirmelerinde deve notasyonu ve fonksiyon isimlendirmelerinde paskal notasyonu kullanılmalıdır.
- 9) Herhangi bir rekürsif fonksiyon kullanımına izin **yoktur.**
- 10) **goto** deyimi kullanılarak program akışının deęiřtirilmesine izin **yoktur.**
- 11) Yazdığınız programın içerisine **MUHAKKAK** açıklayıcı satırlar eklenmelidir.
- 12) Geliřtirme ortamının yazım sitiline müdahale edilmemelidir. Girintilere özellikle dikkat edilmelidir.
- 13) Hazırladığınız uygulamayı **AdSoyad_OgrenciNumarasi_BM104_Proje1.cpp** řeklinde isimlendirmeniz ve son teslim saatine kadar **DBS'ye (Ders Bilgi Sistemi)** yüklemeniz gerekmektedir. (.cpp dosyası haricinde herhangi bir başka dosya ya da sıkıştırılmış bir dosya teslimi kabul **edilmevecektir.**)
- 14) Teslim ededeęiniz proje **SADECE ve SADECE** kendi emeęinizin ürünü olmalıdır. Aksine hareket edenleri řok ciddi yaptırımların bekledięini unutmayınız.
- 15) **A, B, C ve İngilizce öęretim** grubundaki öęrencilerin teslim ettikleri kodların tamamının özel bir yazılım ile benzerlik testine tabi tutulacaęını unutmayınız.
- 16) Proje teslim kurallarına uymayan ve/veya intihal içeren projelerin deęerlendirmeye alınmayacaęını unutmayınız.