Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 4 з дисципліни

«Основи програмування - 2»

«Успадкування та поліморфізм»

Варіант 4

Виконав: Бондаренко М.В. ІП-13

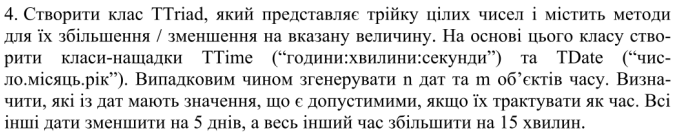
Перевірила: Вечерковська А.С.

Київ 2022

**Лабораторна робота 4**

**Тема:** успадкування та поліморфізм.

**Постановка задачі**

****

**Код C#:**

**Program.cs:**

*using* System;  
*using* System.Collections.Generic;  
  
*namespace* lab4  
{  
 *internal class* Program  
 {  
 *public static void* Main(*string*[] args)  
 {  
 List<TDate> dates = Funcs.DatesGenerate();  
 List<TTime> times = Funcs.TimesGenerate();  
   
 Console.WriteLine("\nInitial dates:");  
 Funcs.ListOutput(dates);  
 Console.WriteLine("\nInitial times:");  
 Funcs.ListOutput(times);  
  
 List<TDate> dates\_like\_time = Funcs.DatesLikeTime(dates);  
 Console.WriteLine("\nDates that can be interpreted like time:");  
 Funcs.ListOutput(dates\_like\_time);  
   
 Funcs.ChangeDates(dates);  
 Funcs.ChangeTimes(times);  
   
 Console.WriteLine("\nChanged dates:");  
 Funcs.ListOutput(dates);  
 Console.WriteLine("\nChanged times:");  
 Funcs.ListOutput(times);  
 }  
 }  
}

**Funcs.cs:**

*using* System;  
*using* System.Collections.Generic;  
  
*namespace* lab4  
{  
 *public class* Funcs  
 {  
 *public static* List<TDate> DatesGenerate()  
 {  
 List<TDate> dates = *new* List<TDate>();  
 Random random = *new* Random();  
 Console.Write("Enter the amount of dates: ");  
 *int* n = Convert.ToInt32(Console.ReadLine());  
 *for* (*int* i = 0; i < n; i++)  
 {  
 *int* day = random.Next(1, 31);  
 *int* month = random.Next(1, 12);  
 *int* year = random.Next(0, 100);  
 TDate date = *new* TDate(day, month, year);  
 dates.Add(date);  
 }  
  
 *return* dates;  
 }  
   
 *public static* List<TTime> TimesGenerate()  
 {  
 List<TTime> times = *new* List<TTime>();  
 Random random = *new* Random();  
 Console.Write("Enter the amount of times: ");  
 *int* n = Convert.ToInt32(Console.ReadLine());  
 *for* (*int* i = 0; i < n; i++)  
 {  
 *int* hours = random.Next(0, 24);  
 *int* minutes = random.Next(0, 60);  
 *int* seconds = random.Next(0, 60);  
 TTime time = *new* TTime(hours, minutes, seconds);  
 times.Add(time);  
 }  
  
 *return* times;  
 }  
  
 *public static void* ListOutput(List<TDate> list)  
 {  
 *foreach* (*var* date *in* list)  
 {  
 date.Output();  
 }  
 }  
  
 *public static void* ListOutput(List<TTime> list)  
 {  
 *foreach* (*var* time *in* list)  
 {  
 time.Output();  
 }  
 }  
  
 *public static bool* IsLikeTime(TDate date)  
 {  
 *if* (date.Number1 < 24 && date.Number3 < 60)  
 {  
 *return true*;  
 }  
  
 *return false*;  
 }  
   
   
 *public static void* ChangeDates(List<TDate> dates)  
 {  
 *foreach* (*var* date *in* dates)  
 {  
 *if* (!IsLikeTime(date))  
 {  
 date.DecreaseNumber1(5);  
 }  
 }  
 }  
   
 *public static void* ChangeTimes(List<TTime> times)  
 {  
 *foreach* (*var* time *in* times)  
 {  
 time.IncreaseNumber2(15);  
 }  
 }  
   
 *public static string* ChangeFormat(*int* n)  
 {  
 *string* n\_str = Convert.ToString(n);  
 *if* (n < 10)  
 {  
 n\_str = "0" + n\_str;  
 }  
  
 *return* n\_str;  
 }  
  
 *public static* List<TDate> DatesLikeTime(List<TDate> dates)  
 {  
 List<TDate> dates\_like\_time = *new* List<TDate>();  
 *foreach* (*var* date *in* dates)  
 {  
 *if* (IsLikeTime(date)) dates\_like\_time.Add(date);  
 }  
  
 *return* dates\_like\_time;  
 }  
 }  
}

**TTriad.cs:**

*namespace* lab4  
{  
 *public abstract class* TTriad  
 {  
 *public int* Number1 { get; set; }  
 *public int* Number2 { get; set; }  
 *public int* Number3 { get; set; }  
   
 *protected* TTriad(*int* n1, *int* n2, *int* n3)  
 {  
 Number1 = n1;  
 Number2 = n2;  
 Number3 = n3;  
 }  
   
 *public abstract void* IncreaseNumber1(*int* n);  
 *public abstract void* DecreaseNumber1(*int* n);  
 *public abstract void* IncreaseNumber2(*int* n);  
 *public abstract void* DecreaseNumber2(*int* n);  
 *public abstract void* IncreaseNumber3(*int* n);  
 *public abstract void* DecreaseNumber3(*int* n);  
 }  
}

**TDate.cs:**

*using* System;  
  
*namespace* lab4  
{  
 *public class* TDate : TTriad  
 {  
 *public* TDate(*int* n1, *int* n2, *int* n3) : *base*(n1, n2, n3)  
 {  
 }  
  
 *public override void* IncreaseNumber1(*int* n)  
 {  
 Number1 += n;  
 *while* (Number1 > 30)  
 {  
 Number1 -= 30;  
 IncreaseNumber2(1);  
 }  
 }  
   
 *public override void* DecreaseNumber1(*int* n)  
 {  
 Number1 -= n;  
 *while* (Number1 <= 0)  
 {  
 Number1 += 30;  
 DecreaseNumber2(1);  
 }  
 }  
   
 *public override void* IncreaseNumber2(*int* n)  
 {  
 Number2 += n;  
 *while* (Number2 > 12)  
 {  
 Number2 -= 12;  
 IncreaseNumber3(1);  
 }  
 }  
   
 *public override void* DecreaseNumber2(*int* n)  
 {  
 Number2 -= n;  
 *while* (Number2 <= 0)  
 {  
 Number2 += 12;  
 DecreaseNumber3(1);  
 }  
 }  
   
 *public override void* IncreaseNumber3(*int* n)  
 {  
 Number3 += n;  
 }  
  
 *public override void* DecreaseNumber3(*int* n)  
 {  
 Number3 -= n;  
 }  
  
 *public void* Output()  
 {  
 Console.WriteLine($"{Funcs.ChangeFormat(Number1)}.{Funcs.ChangeFormat(Number2)}.{Funcs.ChangeFormat(Number3)}");  
 }  
 }  
}

**TTime.cs:**

*using* System;  
  
*namespace* lab4  
{  
 *public class* TTime : TTriad  
 {  
 *public* TTime(*int* n1, *int* n2, *int* n3) : *base*(n1, n2, n3)  
 {  
 }  
  
 *public override void* IncreaseNumber1(*int* n)  
 {  
 Number1 += n;  
 *while* (Number1 >= 24)  
 {  
 Number1 -= 24;  
 }  
 }  
  
 *public override void* DecreaseNumber1(*int* n)  
 {  
 Number1 -= n;  
 *while* (Number1 < 0)  
 {  
 Number1 += 24;  
 }  
 }  
  
 *public override void* IncreaseNumber2(*int* n)  
 {  
 Number2 += n;  
 *while* (Number2 >= 60)  
 {  
 Number2 -= 60;  
 IncreaseNumber1(1);  
 }  
 }  
  
 *public override void* DecreaseNumber2(*int* n)  
 {  
 Number2 -= n;  
 *while* (Number2 < 0)  
 {  
 Number2 += 60;   
 DecreaseNumber1(1);  
 }  
 }  
  
 *public override void* IncreaseNumber3(*int* n)  
 {  
 Number3 += n;  
 *while* (Number3 >= 60)  
 {  
 Number3 -= 60;  
 IncreaseNumber2(1);  
 }  
 }  
  
 *public override void* DecreaseNumber3(*int* n)  
 {  
 Number3 -= n;  
 *while* (Number3 < 0)  
 {  
 Number3 += 60;   
 DecreaseNumber2(1);  
 }  
 }  
  
 *public void* Output()  
 {  
 Console.WriteLine($"{Funcs.ChangeFormat(Number1)}:{Funcs.ChangeFormat(Number2)}:{Funcs.ChangeFormat(Number3)}");  
 }  
 }  
}

**Console:**

Enter the amount of dates: 3

Enter the amount of times: 3

Initial dates:

07.03.53

26.05.42

20.04.57

Initial times:

09:33:02

22:05:29

16:07:56

Dates that can be interpreted like time:

07.03.53

20.04.57

Changed dates:

07.03.53

21.05.42

20.04.57

Changed times:

09:48:02

22:20:29

16:22:56

**Код на Python:**

**main.py:**

import Funcs  
  
dates = Funcs.DatesGenerate()  
times = Funcs.TimesGenerate()  
  
print("\nInitial dates:")  
Funcs.ListOutput(dates)  
print("\nInitial times:")  
Funcs.ListOutput(times)  
  
dates\_like\_time = Funcs.DatesLikeTime(dates)  
print("\nDates that can be interpreted like time:")  
Funcs.ListOutput(dates\_like\_time)  
  
Funcs.ChangeDates(dates)  
Funcs.ChangeTimes(times)  
  
print("\nChanged dates:")  
Funcs.ListOutput(dates)  
print("\nChanged times:")  
Funcs.ListOutput(times)

**TTriad.py:**

from abc import abstractmethod  
  
  
class TTriad:  
 def \_\_init\_\_(self, number1, number2, number3):  
 self.\_\_number1 = number1  
 self.\_\_number2 = number2  
 self.\_\_number3 = number3  
  
 @property  
 def Number1(self):  
 return self.\_\_number1  
  
 @Number1.setter  
 def Number1(self, number1):  
 self.\_\_number1 = number1  
  
 @property  
 def Number2(self):  
 return self.\_\_number2  
  
 @Number2.setter  
 def Number2(self, number2):  
 self.\_\_number2 = number2  
  
 @property  
 def Number3(self):  
 return self.\_\_number3  
  
 @Number3.setter  
 def Number3(self, number3):  
 self.\_\_number3 = number3  
  
  
 @abstractmethod  
 def IncreaseNumber1(self, n):  
 pass  
  
 @abstractmethod  
 def DecreaseNumber1(self, n):  
 pass  
  
 @abstractmethod  
 def IncreaseNumber2(self, n):  
 pass  
  
 @abstractmethod  
 def DecreaseNumber2(self, n):  
 pass  
  
 @abstractmethod  
 def IncreaseNumber3(self, n):  
 pass  
  
 @abstractmethod  
 def DecreaseNumber3(self, n):  
 pass

**TTime.py:**

from TTriad import TTriad  
import Funcs  
  
  
class TTime(TTriad):  
 def \_\_init\_\_(self, number1, number2, number3):  
 super().\_\_init\_\_(number1, number2, number3)  
  
 def IncreaseNumber1(self, n):  
 self.Number1 += n  
 while (self.Number1 >= 24):  
 self.Number1 -= 24  
  
 def DecreaseNumber1(self, n):  
 self.Number1 -= n  
 while self.Number1 < 0:  
 self.Number1 += 24  
  
 def IncreaseNumber2(self, n):  
 self.Number2 += n  
 while self.Number2 >= 60:  
 self.Number2 -= 60  
 self.IncreaseNumber1(1)  
  
 def DecreaseNumber2(self, n):  
 self.Number2 -= n  
 while self.Number2 < 0:  
 self.Number2 += 60  
 self.DecreaseNumber1(1)  
  
 def IncreaseNumber3(self, n):  
 self.Number3 += n  
 while self.Number3 >= 60:  
 self.Number3 -= 60  
 self.IncreaseNumber2(1)  
  
 def DecreaseNumber3(self, n):  
 self.Number3 -= n  
 while self.Number3 < 0:  
 self.Number3 += 60  
 self.DecreaseNumber2(1)  
  
  
 def Output(self):  
 print(  
 f"{Funcs.ChangeFormat(self.Number1)}:{Funcs.ChangeFormat(self.Number2)}:{Funcs.ChangeFormat(self.Number3)}")

**TDate.py:**

import Funcs  
from TTriad import TTriad  
  
  
class TDate(TTriad):  
 def \_\_init\_\_(self, number1, number2, number3):  
 super().\_\_init\_\_(number1, number2, number3)  
  
 def IncreaseNumber1(self, n):  
 self.Number1 += n  
 while self.Number1 > 30:  
 self.Number1 -= 30  
 self.IncreaseNumber2(1)  
  
 def DecreaseNumber1(self, n):  
 self.Number1 -= n  
 while self.Number1 <= 0:  
 self.Number1 += 30  
 self.DecreaseNumber2(1)  
  
  
 def IncreaseNumber2(self, n):  
 self.Number2 += n  
 while self.Number2 > 12:  
 self.Number2 -= 12  
 self.IncreaseNumber3(1)  
  
 def DecreaseNumber2(self, n):  
 self.Number2 -= n  
 while self.Number2 <= 0:  
 self.Number2 += 12  
 self.DecreaseNumber3(1)  
  
  
 def IncreaseNumber3(self, n):  
 self.Number3 += n  
  
 def DecreaseNumber3(self, n):  
 self.Number3 -= n  
  
 def Output(self):  
 print(f"{Funcs.ChangeFormat(self.Number1)}.{Funcs.ChangeFormat(self.Number2)}.{Funcs.ChangeFormat(self.Number3)}")

**Funcs.py:**

import random  
from TDate import TDate  
from TTime import TTime  
  
  
def DatesGenerate():  
 dates = []  
 n = int(input("Enter the amount of dates: "))  
 for i in range(n):  
 day = random.randint(1, 31)  
 month = random.randint(1, 12)  
 year = random.randint(0, 100)  
 date = TDate(day, month, year)  
 dates.append(date)  
 return dates  
  
def TimesGenerate():  
 times = []  
 n = int(input("Enter the amount of times: "))  
 for i in range(n):  
 hours = random.randint(0, 24)  
 minutes = random.randint(0, 60)  
 seconds = random.randint(0, 60)  
 time = TTime(hours, minutes, seconds)  
 times.append(time)  
 return times  
  
def ListOutput(list):  
 for obj in list:  
 obj.Output()  
  
def IsLikeTime(date):  
 if date.Number1 < 24 and date.Number3 < 60:  
 return True  
 return False  
  
def ChangeDates(dates):  
 for date in dates:  
 if not IsLikeTime(date):  
 date.DecreaseNumber1(5)  
  
def ChangeTimes(times):  
 for time in times:  
 time.IncreaseNumber2(15)  
  
def ChangeFormat(n):  
 n\_str = str(n)  
 if n < 10:  
 n\_str = "0" + n\_str  
 return n\_str  
  
def DatesLikeTime(dates):  
 dates\_like\_time = []  
 for date in dates:  
 if IsLikeTime(date):  
 dates\_like\_time.append(date)  
 return dates\_like\_time

**Console:**

Enter the amount of dates: 3

Enter the amount of times: 3

Initial dates:

14.06.13

18.03.36

14.12.61

Initial times:

17:50:41

11:32:16

19:30:30

Dates that can be interpreted like time:

14.06.13

18.03.36

Changed dates:

14.06.13

18.03.36

09.12.61

Changed times:

18:05:41

11:47:16

19:45:30