***გამოცდის ფორმატი***

**\*მონიშნეთ გამოცდის ფორმატი (მიუთითეთ √)**

|  |  |
| --- | --- |
| დახურული წიგნი |  |
| ღია წიგნი | **√** |
| **\*ღია წიგნის შემთხვევაში მონიშნეთ გამოცდაზე ნებადართული ელემენტები (მიუთითეთ √)** | |
| სალექციო მასალები (პრეზენტაცია და სხვა) |  |
| ელექტრონული წიგნები | **√** |
| წიგნები |  |
| კონსპექტები |  |
| ლექსიკონი |  |
| კალკულატორი |  |
| ლეპტოპი/პლანშეტი |  |

**\* გამოცდის ჩატარების წესი იხილეთ ,,დესკტოპზე“ საქაღალდეში Exam materials**

***საგამოცდო საკითხების ფორმა***

***ვარიანტი # 1***

|  |  |  |  |
| --- | --- | --- | --- |
| ***სკოლა*/*საგანმანათლებლო პროგრამა*** | მათემატიკა და კომპიუტერული მეცნიერება | ***სტუდენტის მიერ მიღებული ქულა*** |  |
| ***საგანი*** | პროგრამირების აბსტრაქციები | | |
| ***ლექტორი*** | გ. ბოჭორიშვილი | | |
| ***კურსი*** | I, II | | |
| ***ჯგუფი*** |  | | |
| ***გამოცდის ფორმა*** | ღია წიგნი | | |
| ***გამოცდის***  ***ხანგრძლივობა*** | 1 საათი | | |
| ***მაქსიმალური ქულა*** | 60 | | |
| ***სტუდენტის სახელი და გვარი:*** | | | |

სახელი: ქულა:

პროგრეს ტესტი

პროგრამირების აბსტრაქციებში

**შეასრულეთ შემდეგი ინსტრუქციები, წინააღმდეგ შემთხვევაში შესაძლოა თქვენი ნაშრომი არ შეფასდეს.**

1. **გახსენით project ფოლდერში არსებული პროექტი VS 2008 ში**
2. **თუ Visual Studio ში არ გიჩანთ ფაილები, View მენიუში მონიშნეთ Solution Explorer პუნქტი**
3. **თითოეული ამოცანისთვის დაგხვდებათ შესაბამისი \*.cpp ფაილი, სადაც უნდა დაწეროთ თქვენი ამოხსნა. ფუნქციის პროტოტიპის შეცლა არ შეიძლება. შეგიძლიათ შემოიტანოთ დამხმარე ფუნქციები. არ გაქვთ გლობალური ცვლადების გამოყენების უფება.**
4. **main.cpp ფაილში წერია ამოცანების ტესტებზე შემოწმება. მისი გაშვებით ხდება ამოცანებში მოცემული ფუნქციების გამოძახება სხვადასხვა input-ით და დაბრუნებული პასუხის შემოწმება. კონსოლში გამოვა ინფორმაცია თუ რომელ ტესტებზე გაიარა\ჩაიჭრა თქვენმა ამოხსნამ.**
5. **ამოხსნის დასრულების შემდეგ დაარქივეთ „problems” ფოლდერი და არქივს დაარქვით სახელად თქვენი მეილის პრეფიქსი, მაგალითად gboch10.zip**
6. **ატვირთეთ არქივი კლასრუმზე შესაბამის ტესტზე**

**საყურადღებო:**

1. კოდი გაუშვით f5 ღილაკზე დაჭერით ან მწვანე Play ღილაკზე დაჭერით, წინააღმდეგ შემთხვევაში კონსოლის ფორმა არ გამოჩნდება.
2. Fatal error LNK1168 cannot open - ამ შეცდომის შემთხვევაში ctrl + shift + esc ღილაკებით გახსენით task manager და მოკალით ყველა Blank Project პროცესი
3. თუ კოდში ცვლილება შეგაქვთ მაგრამ გაშვებისას არანაირად არი ისახება, დარწმუნდით რომ წინა წარმატებული ბილდი არ ეშვება. F7 ით ცადეთ დაკომპილირება.
4. თუ გსურთ ხაზების ნუმერაციის გამოჩენა, გახსენით Tools > Options მენიუ, გადადით Text Editor > All Languages > General და მონიშნეთ Line numbers.
5. თუ გაურკვეველი შეცდომა გაქვთ, ცადეთ მენიუდან Build > Clean Solution და Build > Clean Blank Project

**ამოცანა 1. ხურდა (60 ქულა)**

თქვენი ამოცანაა მოლარეს დაეხმაროთ მომხმარებლისთვის ხურდის დაბრუნებაში. დაწერეთ giveChange ფუნქციის რეალიზაცია რომელსაც გადაეცემა სალაროში არსებული კუპიურების სია(cshBills), კლიენტის ხელთ არსებული კუპიურების სია( cstBills) და სერვისის ღირებულება რომლის საფასურსაც იხდის მომხმარებელი. ფუნქციამ უნდა დააბრუნოს true, თუ არსებობს ზუსტი ხურდის დაბრუნების საშუალება და false წინააღმდეგ შემთხვევაში. გაითვალისწინეთ ერთი და იგივე კუპიურა შესაძლოა რამდენიმე ცალი იყოს.

მაგალითი:

cshBills = {3, 4, 3, 3}

cstBills = {5, 5}

price = 3

როგორც ვხედავთ სალაროში არის სამი 3 ლარიანი და ერთ 4 ლარიანი. მომხმარებელს აქვს მხოლოდ ორი 5 ლარინი, ხოლო ნივთი ღირს 3 ლარი.

ამ მაგალითში შესაძლებელია ზურდის ზუსტად დაბრუნება, მომხმარებელმა უნდა მისცეს ორივე 5 ლარიანი და მოლარემ უნდა დაუბრუნოს ერთი 3 ლარინი და ერთი 4 ლარიანი.

bool giveChange(Vector<int> &cshBills, Vector<int> &cstBills, int price);

ჯამურად სალაროში და მომხმარებლის ხელთ არსებული კუპიურების რაოდენობა არ აღემატება 24 ს.

პროგრამა შემოწმდება კომპიუტერზე რომელიც წამში ასრულებს დაახლოებით 109 ოპერაციას. ფუნქციას ერთ ტესტზე არ უნდა ჭირდებოდეს 1 წამზე მეტი დრო, წინააღმდეგ შემთხვევაში ტესტი არ ჩაითვლება.

ტესტების 50% - n < 15

ტესტების 25% - 15 <= n < 20

ტესტების 25% - 20 <= n < 24

main.c ფაილში შეგიძლიათ იხილოთ ტესტები და პასუხები.