Картина, която съдържа текст, графична колекция

Описанието е генерирано автоматичноТЕХНИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА

Факултет по изчислителна техника и автоматизация

Катедра „ КОМПЮТЪРНИ НАУКИ И ТЕХНОЛОГИИ “

**СЕМЕСТРИАЛНА ДОМАШНА РАБОТА**

по дисциплината „Базово програмиране”

на тема: „Монитори”

Вариант 158

|  |  |
| --- | --- |
| Изготвил: Калоян Маринов Енев | Проверил: |
| Специалност: КСТ |  |
| Група: 1 |  |
| Факултетен номер: 21621330 |  |

2021

Съдържание

[I. Задание на проекта 3](#_Toc91106791)

[II. Анализ на решението 4](#_Toc91106792)

[1. Структура за данните в програмата 4](#_Toc91106793)

[2. Реализация на условие A 4](#_Toc91106794)

[3. Реализация на условие B 4](#_Toc91106795)

[4. Реализация на условие C 5](#_Toc91106796)

[5. Реализация на условие D 5](#_Toc91106797)

[6. Реализация на условие E 6](#_Toc91106798)

[7. Реализация на условие F 7](#_Toc91106799)

[8. Реализация на условие G 7](#_Toc91106800)

[9. Реализация на условие H (по желание) 8](#_Toc91106801)

[III. Упътване за употреба 9](#_Toc91106802)

[1. Впишете съответната част от проекта 9](#_Toc91106803)

[2. Впишете съответната част от проекта 9](#_Toc91106804)

[IV. Примерно действие на програмата 10](#_Toc91106805)

[1. Условие A 10](#_Toc91106806)

[2. Условие B 10](#_Toc91106807)

[3. Условие C 10](#_Toc91106808)

[4. Условие D 10](#_Toc91106809)

[5. Условие E 10](#_Toc91106810)

[6. Условие F 11](#_Toc91106811)

[7. Условие G 11](#_Toc91106812)

[8. Условие H (по желание) 11](#_Toc91106813)

Задание на проекта

Да се напише компютърна програма, реализираща информационна система за продажба на монитори (сериен номер, марка, модел, цвят, цена, диагонал на екрана, разделителна способност, вграден TV тунер(да/не), цвят, тип: нов, втора употреба, статус: в продажба, капариран, продаден; в промоция (указва се % на отстъпка – като 0% се счита, че не е в промоция!)). Програмата да съхранява и да обработва данни за наличните монитори. Максималният брой монитори, които могат да се поддържат, е 100.

Базова задача – сложност ниска

A. Меню с избор на функциите в програмата (7 седмица)

B. Добавяне на нови монитори (7-8 седмица)

a. Добавяне на един нов монитор

b. Добавяне на списък от монитори. Въвежда се цяло число n и след него n на брой

нови монитори.

C. Извеждане на всички монитори на екрана (8-9 седмица)

a. Извеждане на монитори с най-голяма разделителна способност

b. Извеждане на монитори втора употреба

Допълнение първо – сложност средна (+ базова задача)

D. Корекция на данни за монитори.

a. Въвеждат се марка и модел и данните за корекция

b. Въвежда се сериен/ни номер/а и се прави корекция на тези монитори

c. Определяне на промоционални %ти за модел монитори и/или за определени

бройки монитори (по сериен номер)

Допълнение второ – сложност висока (+ базова задача + допълнение първо)

E. Продажба на монитори:

a. Въвеждат се модел и марка и се избира от наличните монитори по сериен номер.

b. Капариране на монитор. След капариране се намалява реалната цена с капарото.

Допълнение трето – сложност висока (+ базова задача + допълнение първо + допълнение

второ)

F. Одит на мониторите в наличност в под меню (11-12 седмица.)

a. Извеждане на монитори в подреден ред по сериен номер

b. Извеждане на монитори по критерии:

i. марка, сортирани по цена

ii.всички монитори в промоция, сортирани по цена в намаляващ ред и

визуализиране на новата цена, при прилагане на отстъпката

iii. всички монитори, които имат диагонал на екрана по-голям от

зададен, разделителна способност по-голяма от зададена и вграден TV

тунер, сортирани по марка

G. Данните в програмата да могат да се запазват във файл между две стартирания на

програмата.

Допълнение четвърто – (за допълнителни точки)

H. Допълнителни условия:

a. за точка А: при добавянето на нови телефони, ако в системата има вече въведена

информация, извежда подходящо съобщение

b. Когато се прави корекция на модел монитори, тя важи за всички налични бройки,

освен продадените.

c. За точка B: да се реализира добавяне със запитване за нов запис и прекъсване на

въвеждането.

d. Одита на монитори се прилага само към текущо продаваните монитори.

Анализ на решението

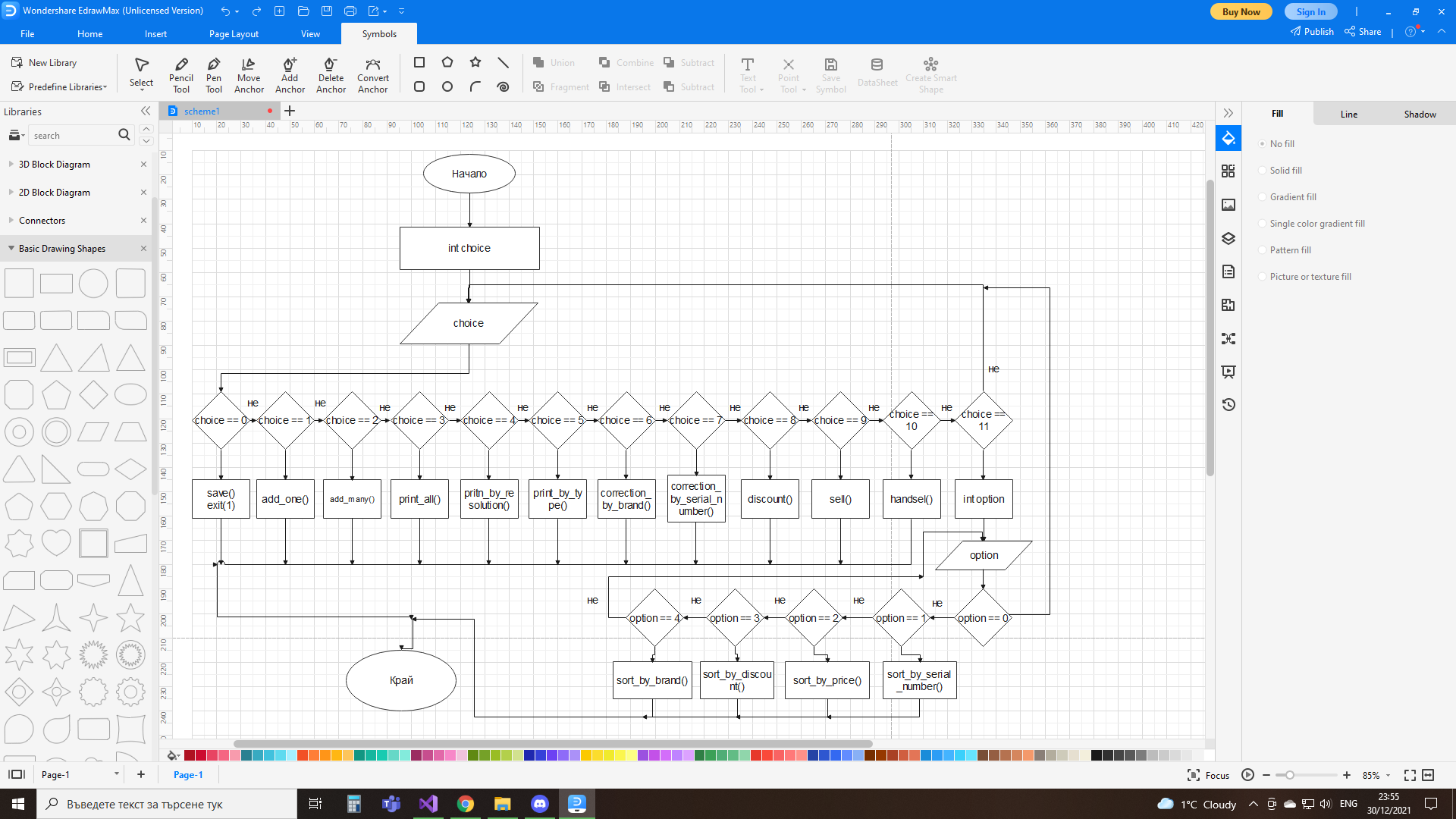
Структура за данните в програмата

|  |  |  |
| --- | --- | --- |
| Структура | Обяснение | Примерени стойности |
| struct monitor  {   1. int serial\_number 2. int diagonal 3. string brand 4. string model 5. string color 6. string tuner 7. string type 8. string status 9. string resolution 10. double price 11. int discount   }; | Характеристики на монитор:   1. Сериен номер 2. Дължина на диагонала 3. Марка 4. Модел 5. Цвят 6. Дали има или няма вграден тунер 7. Тип(Нов или втора употреба) 8. Статус(в продажба, в промоция, капариран, продаден 9. Разделителна способност 10. Цена 11. Ако е в промоция, колко процента е тази промоция | 1. 1001 2. 15 3. LG 4. L10 5. черен 6. да 7. нов 8. в продажба 9. 1920x1080 10. 349.99 11. 20 |

Реализация на условие A

Анализ на алгоритъма, който трябва да се реализира

В main се дефинира променлива choice от тип int. След това се използва цикъл do…while, като е зададено да се изпълнява безкрайно(чрез условие true) и чрез оператор cout и стандартно изходно устройство << се изкарват възможностите на менюто, след което чрез operator cin и стандартно входно устройство >> , се въвежда стойност на променливата choice(от 0 до 11). Чрез функция switch (case) и въведената стойност на choice, се избира избор от показаните на екрана, като с избор 0 се прекъсва цикъла и се излиза от програмата.

Блок схема на алгоритъма  


Функция с която е реализиран алгоритъма

Менюто е в main функцията. Въвежда се избор от 0 до 11 и това води към следващата функция, която е избрана.

#### Входни данни на функцията

Входните данни са променлива от тип int, която представлява избор от възможностите на менюто. Избран е тип int, защото е необходимо цяло число.

#### Изходни данни на функцията или данни, които се извеждат

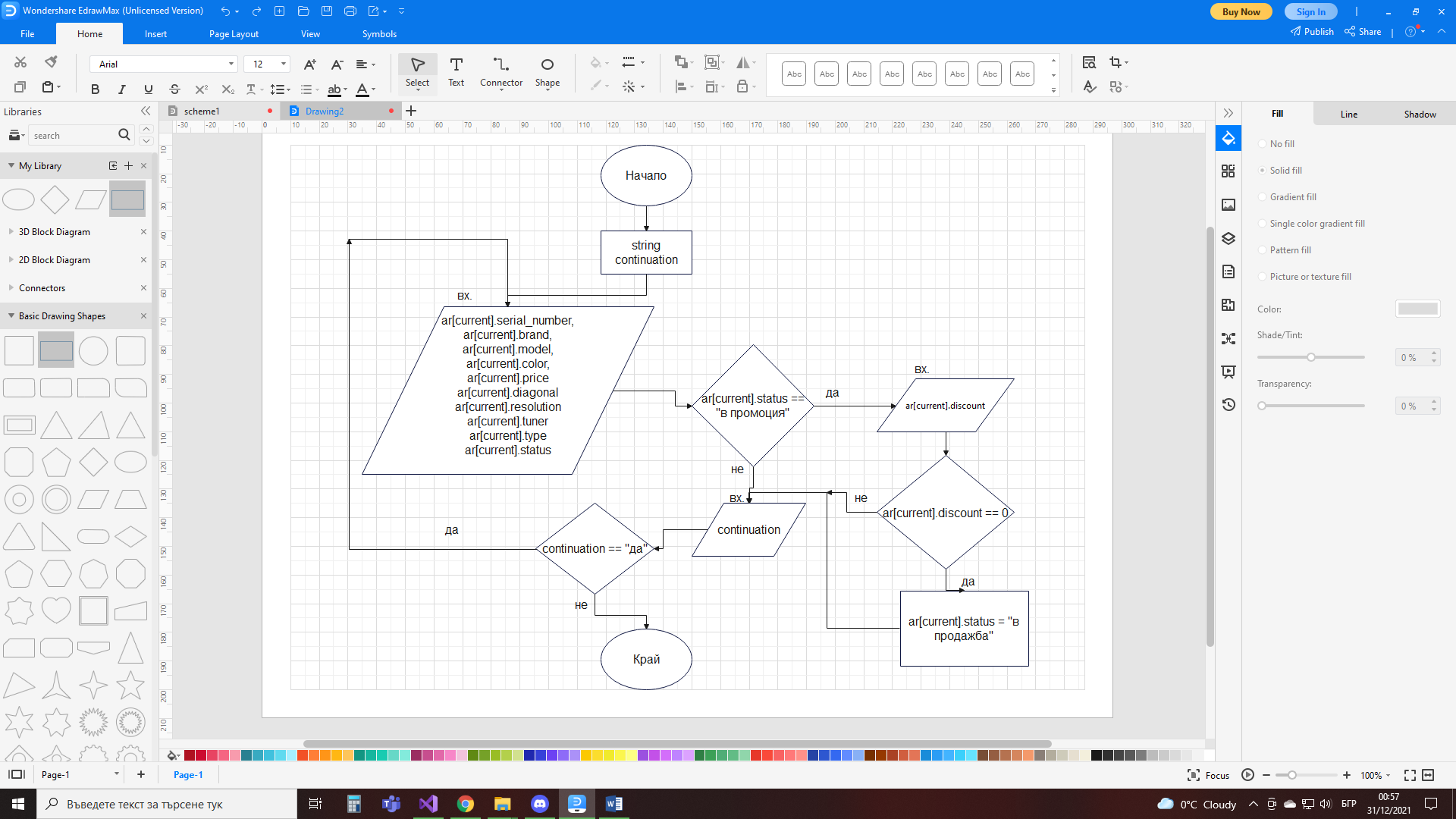
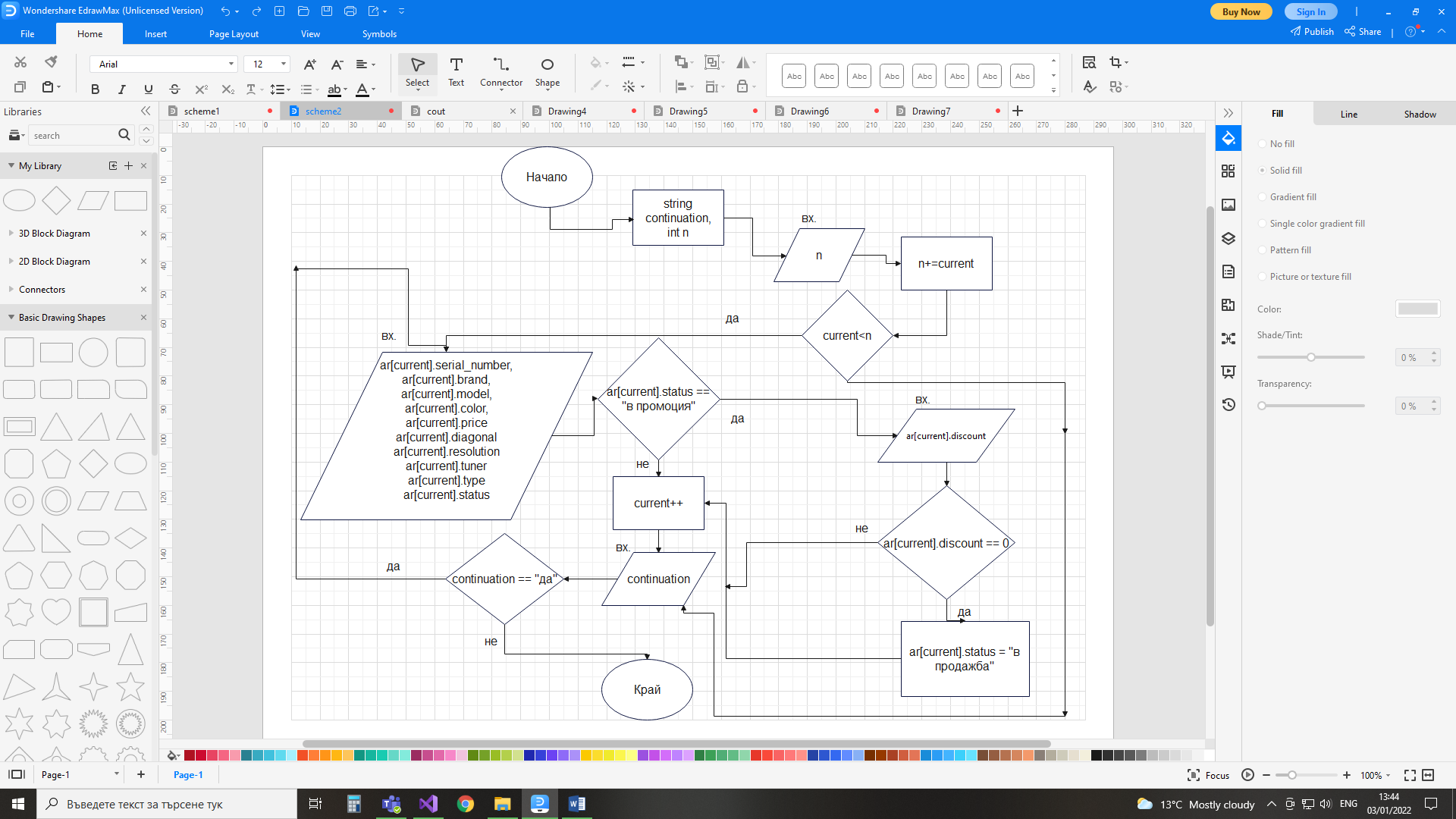
Извеждат се възможностите за избор на менюто.

Реализация на условие B

Анализ на алгоритъма, който трябва да се реализира

3.1.а. При избор 1 от менюто се въвеждат поотделно данните за един нов монитор, който бива добавен в системата.

3.1.b. При избор 2 от менюто се избира колко нови монитора да бъдат добавени чрез променлива n и се въвеждат поотделно данните за всеки един от тях и биват добавени в системата. Въвеждането на монитори се осъществява с цикъл for, който започва от броя на въведените досега монитори, след всяко завъртане, те се увеличават с 1, и приключва като броят им достигне числото на въведената променлива n + броя на мониторите преди началото на функцията.

Блок схема на алгоритъма  
a.  
  
b.  


Функция с която е реализиран алгоритъма

Функциите са от тип void, защото не е необходим резултат.

3.3.a. Функция add\_one(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 1-ва опция в менюто. След приключване програмата се връща към главното меню.

3.3.b. Функция add\_many(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 2-ра опция в менюто. След приключване програмата се връща към главното меню.

#### Входни данни на функцията Въвеждат се поотделно данни за всяка една от променливите в структурата, съответстващи на типа на променливата. За повечето се използва стандартно входно устройство >> и оператор cin, но за някои (status, type), се използва функция getline(), защото понякога изискват интервал. Променлива discount зависи от стойността на друга променлива (status), и като такава се използва оператор if, който решава дали е необходимо да се въведе стойност и за нея или да си остане 0. В подточка b се въвежда, чрез cin и >>, стойност за променлива n, която решава колко пъти да се върти цикъла и колко нови монитора да бъдат въведени.

#### Изходни данни на функцията или данни, които се извеждат Тези функции не връщат резултат. Преди въвеждането на всяка една от данните, се изписва в конзолата неиното наименование и след въвеждане, се извежда следващото. В подточка b в началото се запитва колко нови монитора да бъдат въведени.

Реализация на условие C

Анализ на алгоритъма, който трябва да се реализира

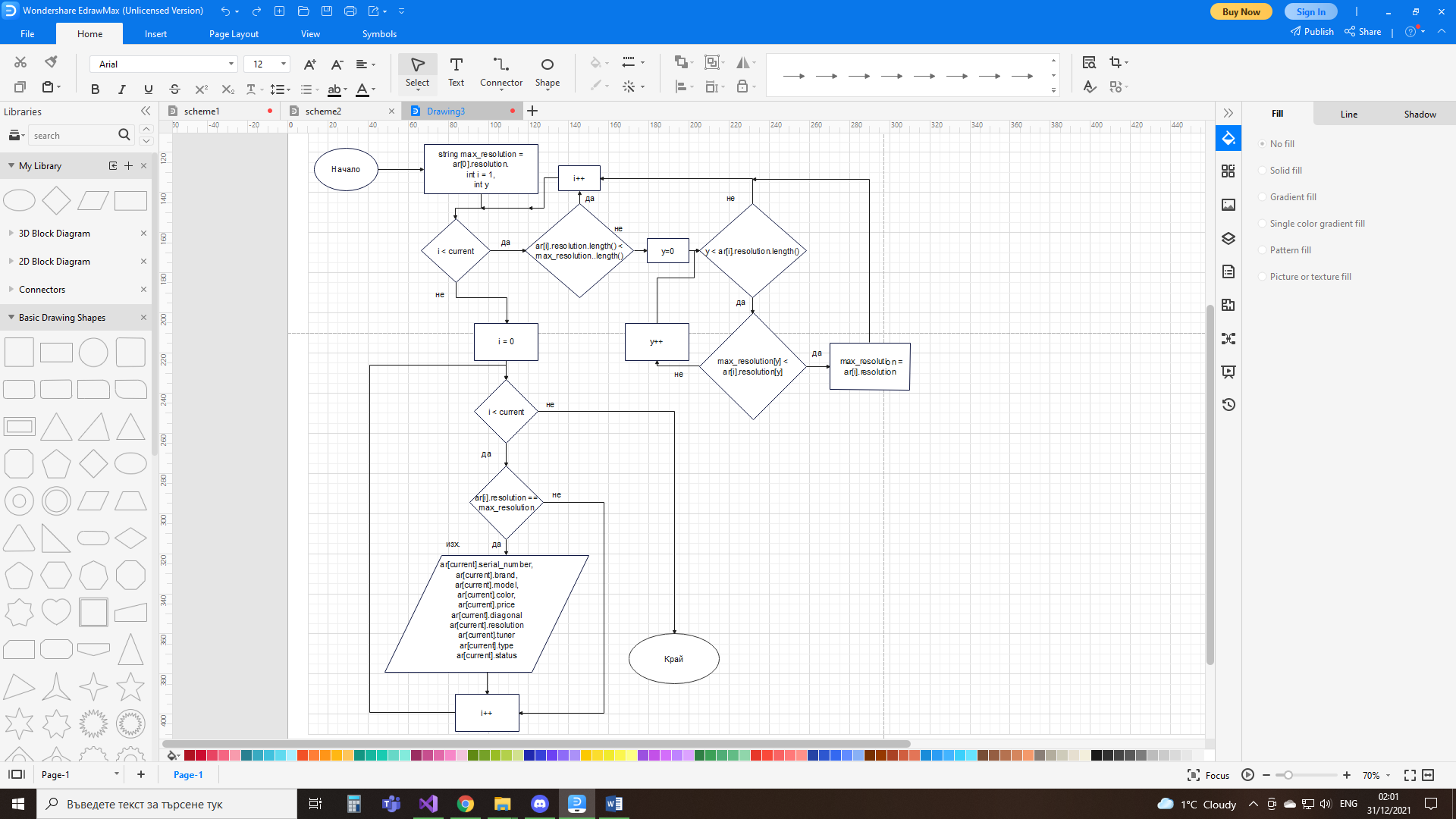
4.1.а. При избор 3 от менюто се извеждат всички данни на досега въведените монитори. Това се осъществява с цикъл.

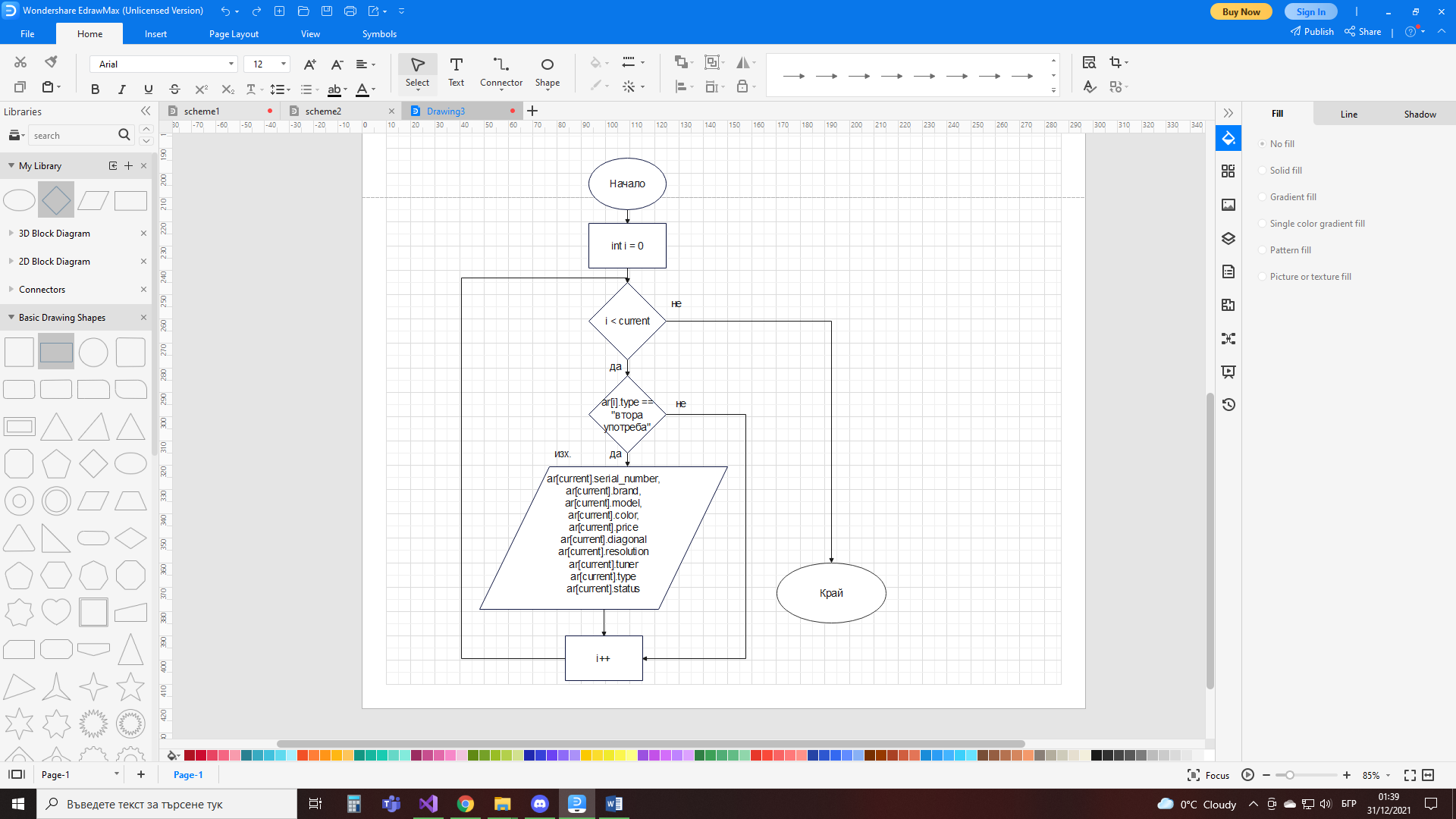
4.1.b. При избор 4 от менюто се извеждат всички данни на мониторите с най-голяма разделителна способност от досега въведените монитори. Това се осъществява с цикъл и оператор if, който проверява дали мониторът е един от тези с най-голяма резолюция. Най-голямата резолюция се определя от друг цикъл, които проверява и сравнява всяка една от досега въведените резолюции и запазва в променлива най-голямата.

4.1.c. При избор 5 от менюто се извеждат всички монитори, които са втора употреба от досега въведените. Това се осъществява с цикъл и оператор if, който проверява дали мониторът е втора употреба.

Блок схема на алгоритъма

a.  
 

b.   


c.  
 

Функция с която е реализиран алгоритъма  
Функциите са от тип void, защото не е необходим резултат.

4.3.а. Функция print\_all(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 3-та опция в менюто. След извеждането се връща към главното меню.

4.3.b. Функция print\_by\_resolution(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 4-та опция в менюто. След извеждането се връща към главното меню.

4.3.c. Функция print\_by\_type(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 5-та опция в менюто. След извеждането се връща към главното меню.

#### Входни данни на функцията Няма входни данни.

#### Изходни данни на функцията или данни, които се извеждат Функциите са от тип void, защото не е необходим резултат. В цикъл for, чиято променлива i започва от 0, увеличава се при всяко завъртане с 1, и приключва когато достигне броя на въведените монитори, се извеждат данните на всеки един монитор чрез cout<< и преди всяка една от конкретните данни се пояснява каква е тя.

Реализация на условие D

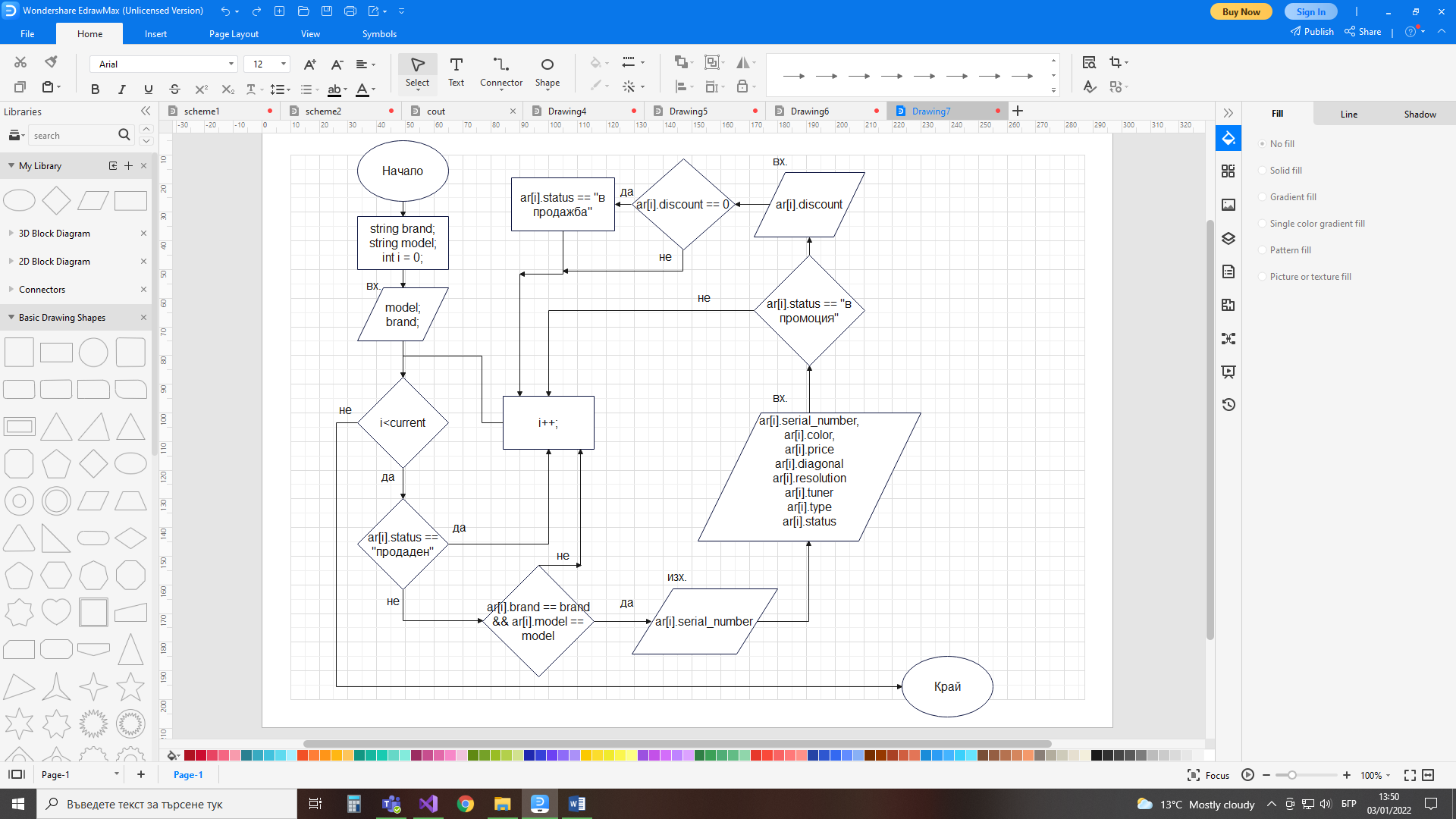
Анализ на алгоритъма, който трябва да се реализира

5.1.а. При избор 6 от менюто се коригират данните на определена марка и модел монитори. Това се осъществява като се въведат от потребителя две нови променливи от тип string (модел и марка). С помощта на цикъл for, ако те съответстват на марката и модела на вече въведен монитор, се извежда неговия сериен номер и се подават една по една всички останали данни, за да им бъдат въведени от потребителя нови стойности.

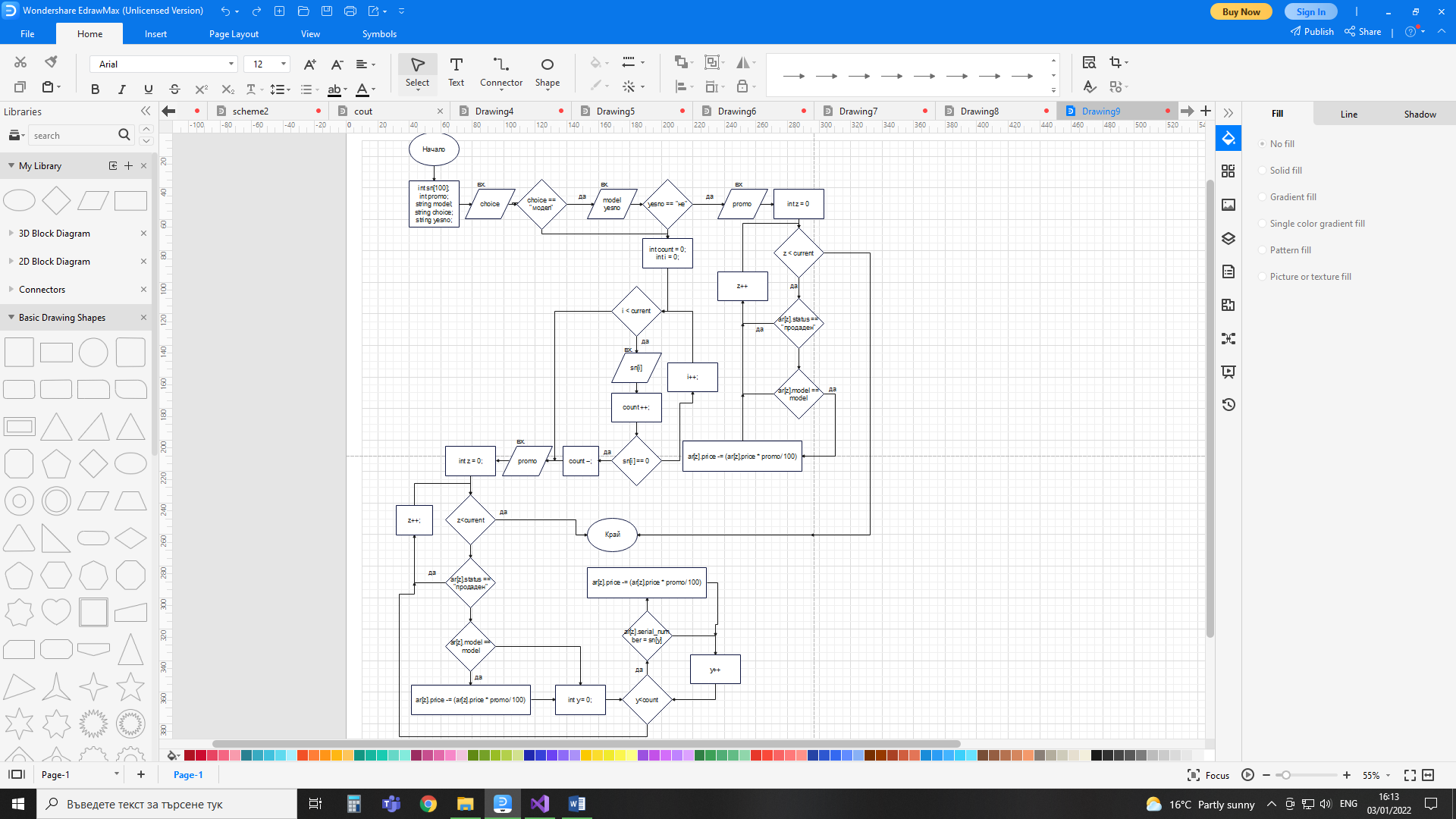
5.1.b. При избор 7 от менюто се коригират данните на един или повече монитора по сериен номер. За това се използва цикъл, който дава възможност на потребителя да избере един или повече серийни номера, които се запазват в масив, докато не въведе 0, с което започва да въвежда за всеки един от въведените серийни номера, всички останали данни на съответните монитори. Коригирането се осъществява с цикъл, който преминава през елементите на локалния за функцията масив и е вложен в друг цикъл, който преминава през елементите на масива, който съдържа данните на мониторите. Във вложения цикъл се проверява дали серийните номера съответстват и ако е да, се въвеждат всички данни, с изключение на серийния номер на съответния монитор.

5.1.c. При избор 8 от менюто се извършва промоция на монитор по модел и/или сериен/ни номер/а. Първо се избира по модел или по сериен номер. Ако е по модел, се избира модел и потребителя се запитва дали желае и по сериен номер. Ако избере „да“, въвежда един или повече серийни номера, сходно на предишната подточка. След това се запитва колко процента да бъде промоцията за избраните монитори. Накрая с цикъл for, който преминава през всеки елемент от масива, и чрез if проверява дали модела и серийния номер съответстват, се осъществява промяна на цената на всеки един от избраните монитори с толкова процента, колкото са зададени.

Блок схема на алгоритъма

а.  
 

b.  


c.  


Функция с която е реализиран алгоритъма  
Функциите са от тип void, защото не е необходим резултат.

5.3.а. Функция correction\_by\_brand(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 6-та опция в менюто. След приключване се връща към главното меню.

5.3.b. Функция correction\_by\_serial\_number(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 7-ма опция в менюто.. След приключване се връща към главното меню.

5.3.c. Функция discount(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 8-ма опция в менюто. След приключване се връща към главното меню.

#### Входни данни на функцията 5.3.1.а. Въвеждат се веднъж марка и веднъж модел на монитори. И след това корекциите, колкото пъти е необходимо. 5.3.1.b. Въвеждат се серийни номера от тип int, колкото потребителя избере(максимум 100, колкото монитора може да има) и прекъсва въвеждането като въведе 0. 5.3.1.c. Въвежда се модел от тип string и/или серийни номера от тип int, сходно на подточка b.

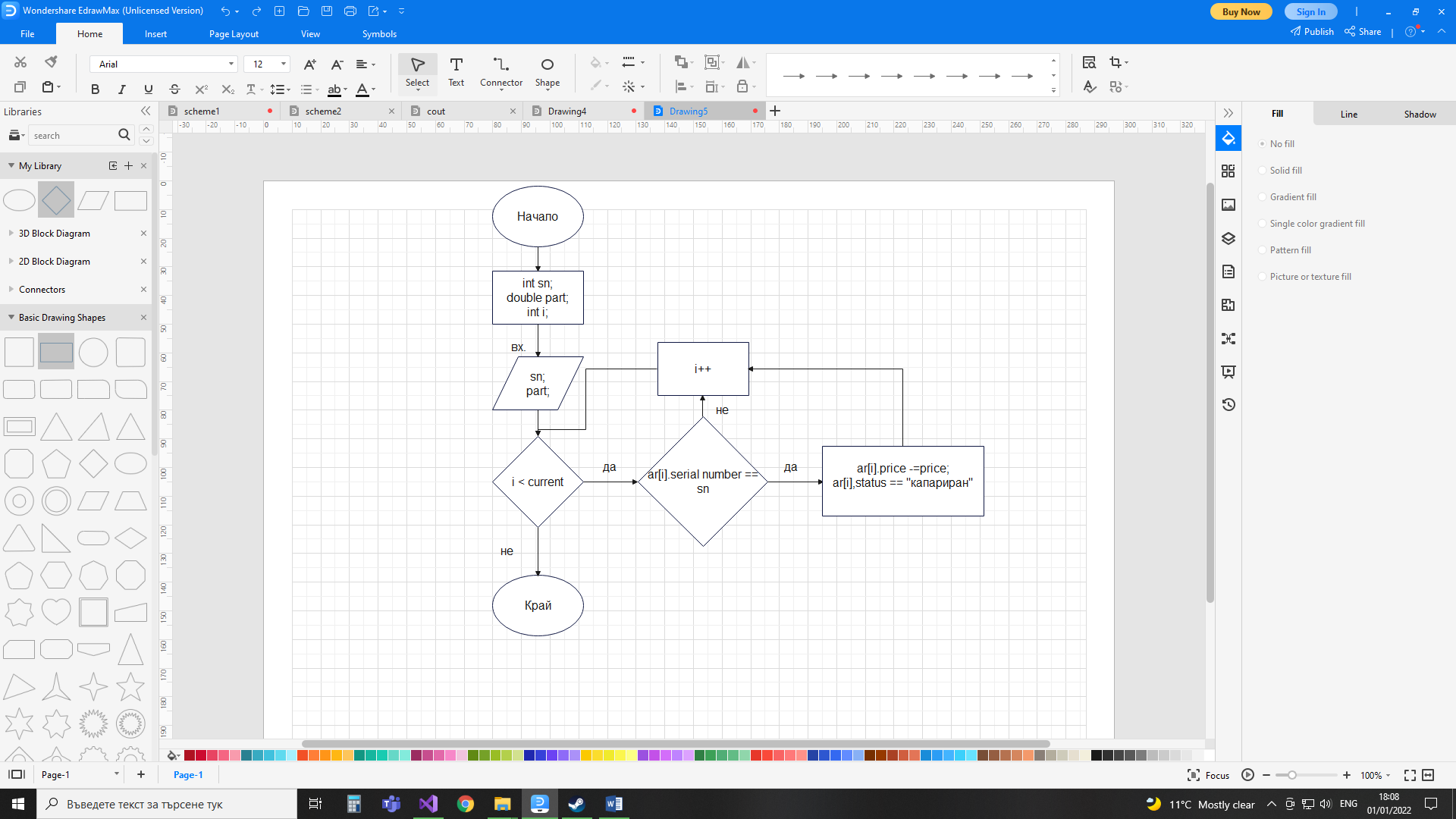
#### Изходни данни на функцията или данни, които се извеждат Извеждат се наименованията на данните, които се изискват за въвеждане от потребителя, преди да бъдат въведени, за да извърши корекциите.

Реализация на условие E

Анализ на алгоритъма, който трябва да се реализира  
6.1.а. След въвеждане на марка и модел, чрез цикъл for и оператор if, се извеждат всички монитори, които съответстват на въведеното. Ако няма такива, се извежда подходящо съобщение. Ако има, се въвежда сериен номер, с цикъл for и оператор if се проверява дали съответства на някой от изведените и ако е да, статуса на съответния монитор става „продаден“.  
6.1.b. Въвежда се сериен номер от потребителя и сума за капариране. След което с цикъл for и оператор if се намира монитора с такъв сериен номер и от неговата цена се изважда капарото, а статуса му става „капариран“.

Блок схема на алгоритъма

a.  


b.  
 

Функция с която е реализиран алгоритъма  
Функциите са от тип void, защото не е необходим резултат.  
6.3.а. Функция sell(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 9-та опция в менюто. След приключване се връща към главното меню.  
6.3.b. Функция handsel(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() като 10-та опция в менюто. След приключване се връща към главното меню.

#### Входни данни на функцията 6.3.1.а. Въвежда се марка от тип string, модел от тип string и ако те съответстват, сериен номер от тип int. 6.3.1.b. Въвежда се сериен номер от тип int и сума на капарото от тип double.

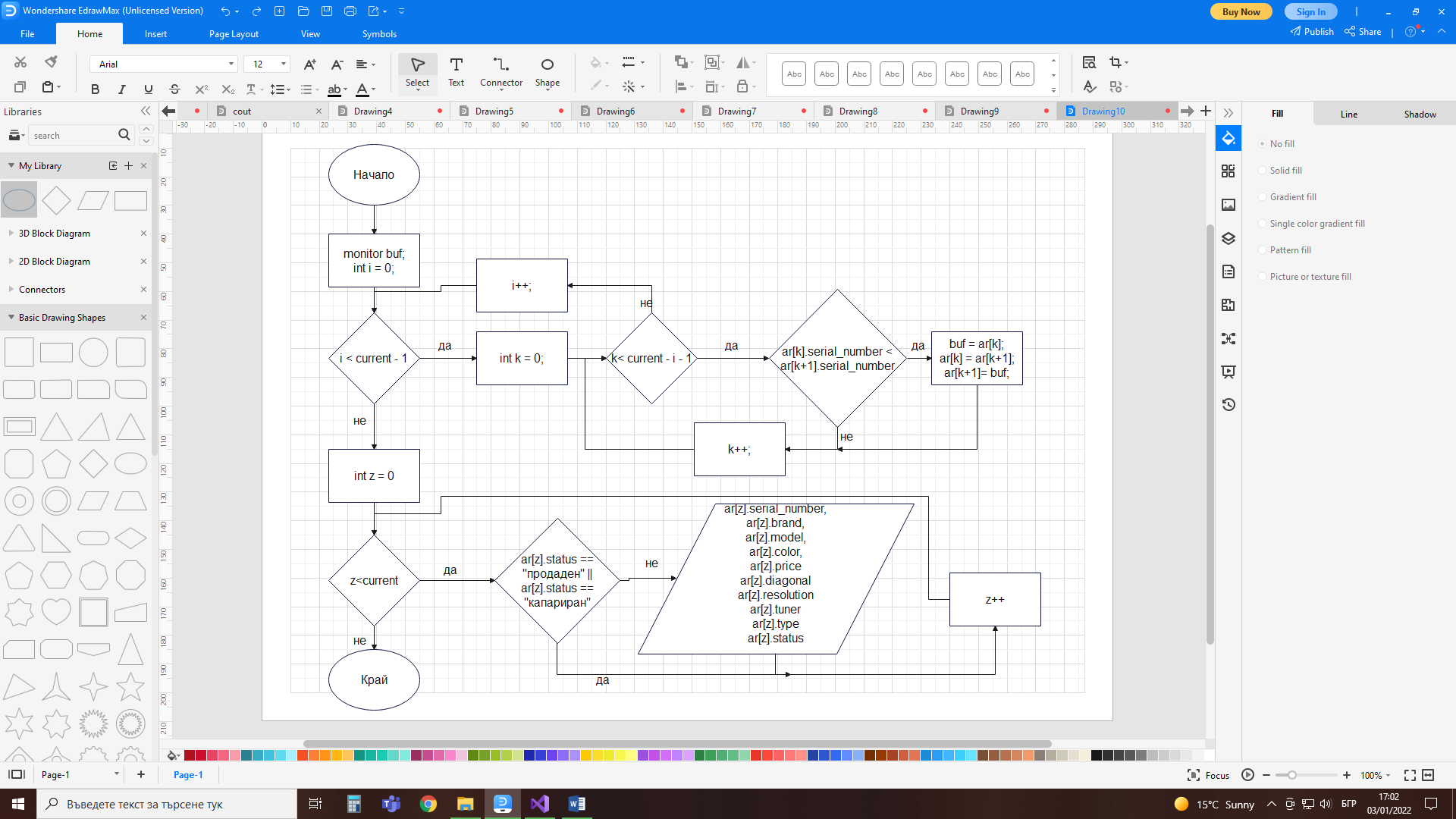
#### Изходни данни на функцията или данни, които се извеждат 6.3.2.а. Извеждат се наименованията на данните, които се изискват за въвеждане от потребителя, преди да бъдат въведени, за да извърши продажбата.

#### 6.3.2.b. Извеждат се наименованията на данните, които се изискват за въвеждане от потребителя, преди да бъдат въведени, за да извърши капарирането.

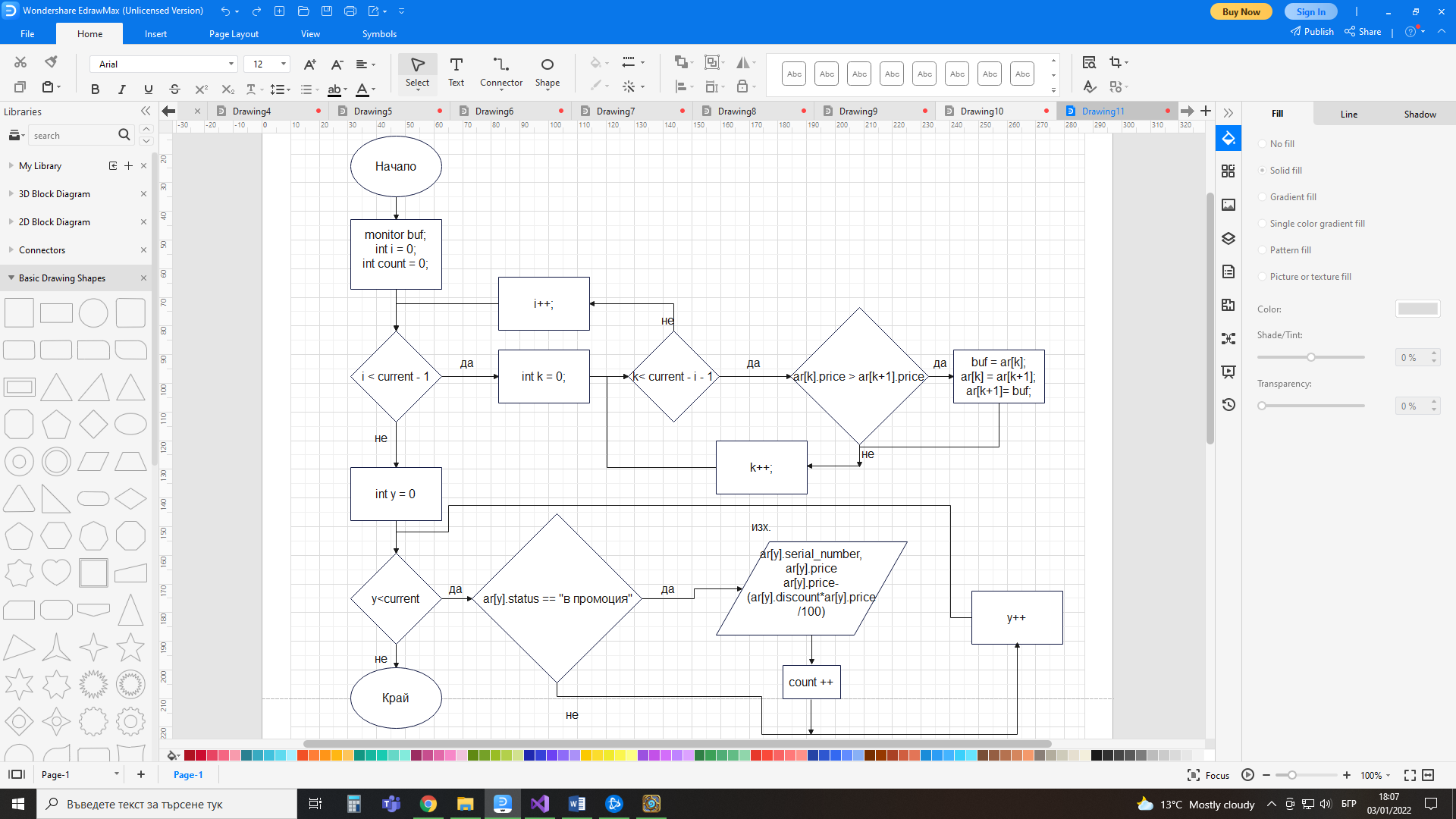
Реализация на условие F

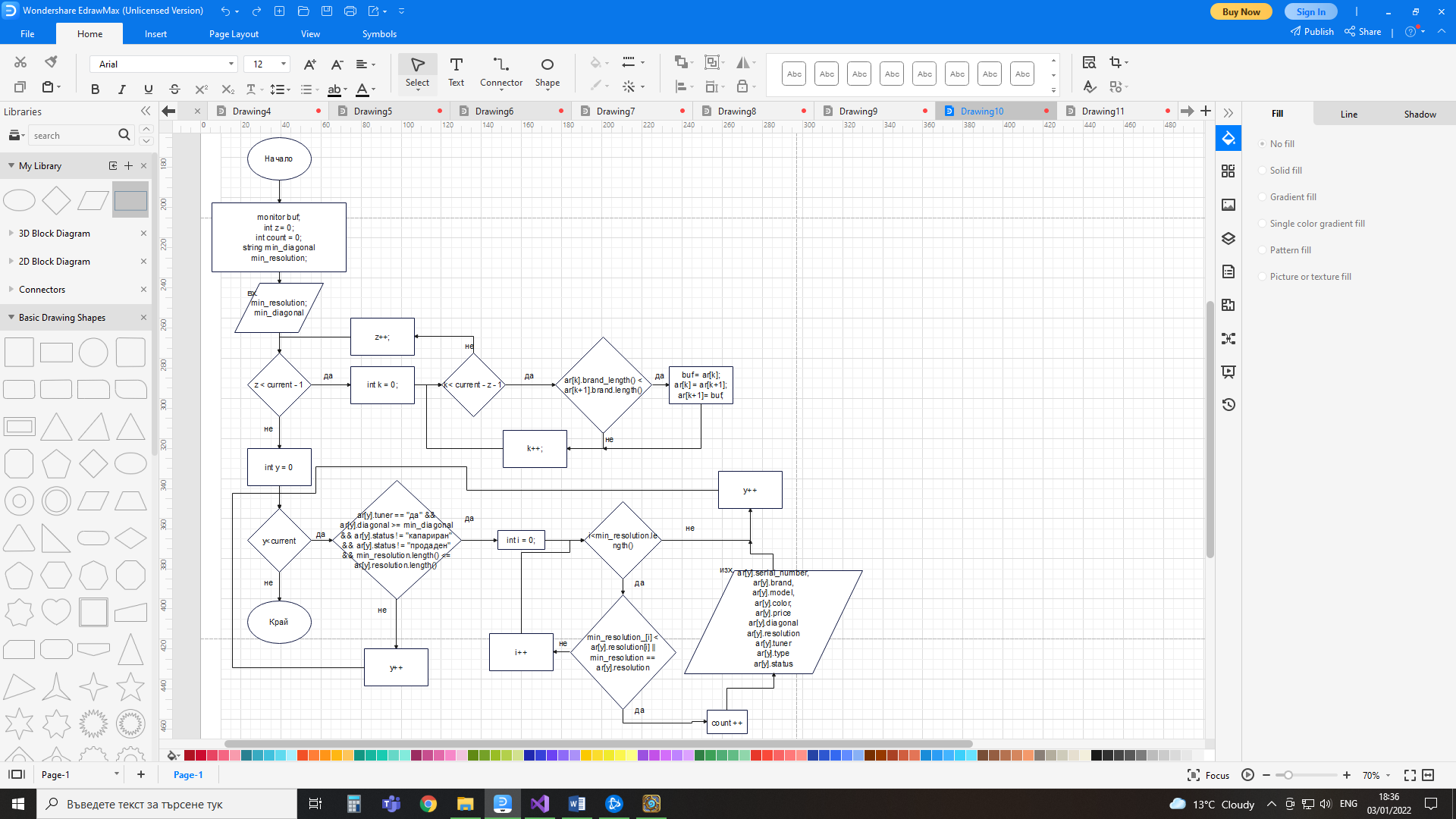
Анализ на алгоритъма, който трябва да се реализира  
7.1.a. Използва се метод на мехурчето за да се сортират вече въведените монитори по сериен номер, след което се извеждат всички монитори.  
7.1.b. Въвежда се марка. Използва се метод на мехурчето за да се сортират вече въведените монитори по цена, след което се извеждат всички монитори, които отговарят на въведената марка. Ако няма съответстващи монитори на тази марка, се извежда подходящо съобщение.  
7.1.c. Използва се метод на мехурчето за да се сортират мониторите по цена, след което чрез оператор if се проверява дали са в промоция и чрез цикъл for, се извеждат мониторите, които са в промоция, старата им цена и цената им след намалението. Ако няма монитори в промоция се извежда подходящо съобщение.  
7.1.d. Въвежда се минимален диагонал, минимална резолюция и с метод на мехурчето се сортират мониторите по марка. След това с оператор if се проверява дали имат тунер, дали имат по-големи диагонали и по-голяма резолюция от минималните и с цикъл for, се извеждат мониторите, които изпълняват тези условия.

Блок схема на алгоритъма

a.  
 

b.  
 

c.  
 

d.  
 

Функция с която е реализиран алгоритъма  
7.3.а. Функция sort\_by\_serial\_number(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main(), в 11-та опция в менюто, като 1-ва в подменюто. След приключване се връща към главното меню.

7.3.b. Функция sort\_by\_price(),която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main(), в 11-та опция, като 2-ра опция в подменюто. След приключване се връща към главното меню.

7.3.c. Функция sort\_by\_discount(),която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main(), в 11-та опция в менюто, като 3-та в подменюто. След приключване се връща към главното меню.

7.3.d. Функция sort\_by\_brand(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main(), в 11-та опция в менюто, като 4-та в подменюто. След приключване се връща към главното меню.

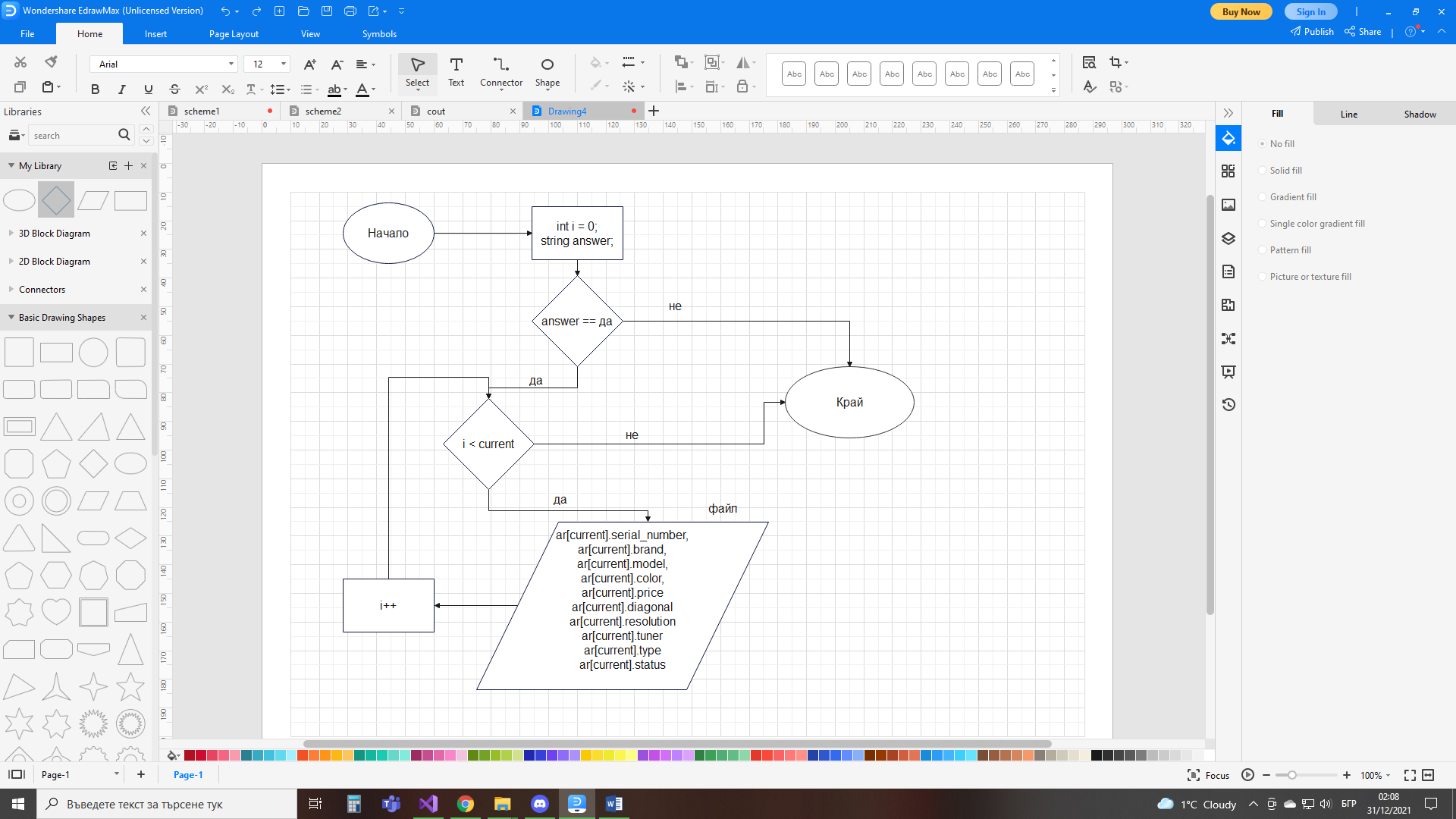
#### Входни данни на функцията 7.3.1.а. Няма входни данни. 7.3.1.b. Въвежда се марка от тип string. 7.3.1.c. Няма входни данни. 7.3.1.d. Въвежда се минимална резолюция от тип string и минимален диагонал от тип int.

#### Изходни данни на функцията или данни, които се извеждат Функците са от тип void, защото не изискват резултат. 7.3.1.a. Извеждат се данните на мониторите след сортиране чрез цикъл for. 7.3.1.b. Извеждат се наименованията на данните, които се изискват за въвеждане от потребителя, преди да бъдат въведени, за да извърши одита. Извеждат се и данните на мониторите след сортиране чрез цикъл for. Или ако няма такива, които отговарят на условията, се извежда подходящо съобщение. 7.3.1.c. Извеждат се и данните на мониторите след сортиране чрез цикъл for. Или ако няма такива, които отговарят на условията, се извежда подходящо съобщение. 7.3.1.d. Извеждат се наименованията на данните, които се изискват за въвеждане от потребителя, преди да бъдат въведени, за да извърши одита. Извеждат се и данните на мониторите след сортиране чрез цикъл for.

Реализация на условие G

Анализ на алгоритъма, който трябва да се реализира  
След въпрос към потребителя, да се запишат ли данните във файл, ако отговорът е „да“, се влиза в цикъл, чрез който данните, записани в главния масив , се прехвърлят във файл.

Блок схема на алгоритъма



Функция с която е реализиран алгоритъма  
Функция save(), която има за параметри масив от типа на структурата, който пази данните и променлива от тип int, която следи броя на вече въведените монитори. Въвежда се в main() при опцията за изход в менюто. След приключване се излиза от програмата.

#### Входни данни на функцията Въвежда се отговор на въпрос – да/не.

#### Изходни данни на функцията или данни, които се извеждат Функцията е от тип void, защото не изисква резултат. Извежда данни в отделен текстов файл.

Реализация на условие H

Анализ на алгоритъма, който трябва да се реализира  
9.1.a. Преди въвеждането на всеки монитор в масива се изписва в конзолата кой монитор е подред от 100-те възможни.  
9.1.b. В циклите във функциите за корекция на монитори се слага оператор if, който проверява статуса на всеки монитор. Ако е „продаден“, чрез команда continue, се продължава към следващата итерация на цикъла.  
9.1.c. В функциите за добавяне на монитори към масива, след всеки добавен монитор, потребителя бива запитан дали иска да продължи да добавя. Ако е да, продължава. Ако е не, се излиза от функцията.  
9.1.d. В циклите във функциите за одит на монитори, се слага оператор if, който проверява статуса на всеки монитор. Ако той е „капариран“ или „продаден“, функцията не извежда мониторите с такъв статус.

#### Входни данни на функцията 9.1.1.а. Няма входни данни. 9.1.1.b. Няма входни данни. 9.1.1.c. Дава се отговор на запитването – да или не. 9.1.1.d. Няма входни данни.

#### Изходни данни на функцията или данни, които се извеждат. 9.1.2.a. Извежда се номера от 1 до 100 на текущия монитор, който предстои да бъде добавен. 9.1.2.c. Извежда се въпрос към потребителя, дали иска да продължи да въвежда нови монитори.

Упътване за употреба

Представете какво трябва да се въведе за правилното използване на програмата, какво се извежда в резултат на нея. Какви са очакваните резултати и какво ще се случи, ако се въведат невалидни данни. Изтрийте този текст, след като се запознаете с него.

Главно меню

Очаквани входни данни  
Число от 0 до 11.

Очакван резултат от изпълнението на конкретния фрагмент от проекта  
Ако бъде въведено нещо различно от интервала [0;11], ще се върне менюто и ще попита за ново въвеждане. Ако са въведени правилни данни, ще приведе програмата към избраната функция.

Добавяне/ коригиране на монитор

Очаквани входни данни  
Серийният номер, диагонала на екрана и процентът на промоция трябва да са цели числа; цената може да е реално число; марката, моделът, цветът са данни от тип string, вграденият тунер е със стойност или да, или не; типът е или нов, или втора употреба; а статусът е едно от четирите: в промоция, в продажба, капариран или продаден; разделителната способност трябва да са две числа, разделени с “x”. При запитване към потребителя дали иска да продължи въвеждането, той трябва да отговори с „да“ или с „не“.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Данните на масива се променят или като се добави нов елемент, или като се коригират части от вече съществуващ елемент. Ако продуктът се въведе „в промоция“, но процентите намаление са 0, той се променя на „в продажба“;

## Продажба на монитор

Очаквани входни данни

Въвеждат се марка и модел от тип string, и сериен номер от тип int.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

След въвеждане на марка и модел, се извеждат всички монитори, които съответстват на тях. Ако няма такива се извежда съобщение „Невалидни данни“. След въвеждане на правилен сериен номер се променя статуса на елемента с такъв сериен номер.

## Капариране на монитор

Очаквани входни данни

Въвеждат се сериен номер от тип int и сума, с която да бъде капариран от тип double.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Изважда се капарото от цената на елемента и статуса му се променя на „капариран“.

## Подменю

Очаквани входни данни

Число от 0 до 4.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Ако бъде въведено нещо различно от интервала [0;4], ще се върне подменюто и ще попита за ново въвеждане. Ако са въведени правилни данни, ще приведе програмата към избраната функция.

## Сортиране по цена

Очаквани входни данни

Въвежда се марка от тип string.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Ако няма съответстваща марка, се извежда съобщение „Няма монитори с такава марка“. Ако има, се извеждат всички монитори с тази марка, сортирани по цена.

## Сортиране по промоция

Очаквани входни данни

Няма входни данни.

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Ако няма монитори в промоция се извежда подходящо съобщение. Ако има, се извеждат те, сортирани по цена, заедно със серийния им номер и новата им цена.

## Сортиране по марка

Очаквани входни данни

Въвежда се минимална резолюция и минимален диагонал, съответно от тип string и int .

Очакван резултат от изпълнението на конкретния фрагмент от проекта

Ако няма съответстващи монитори, се извежда съобщение „Няма такива монитори.“. Ако има, се извеждат тези монитори, сортирани по марка.

## Запис във файл

Очаквани входни данни

При избор 0 от главното меню се запитва дали потребителя иска запис. Трябва да отговори с да или с не.

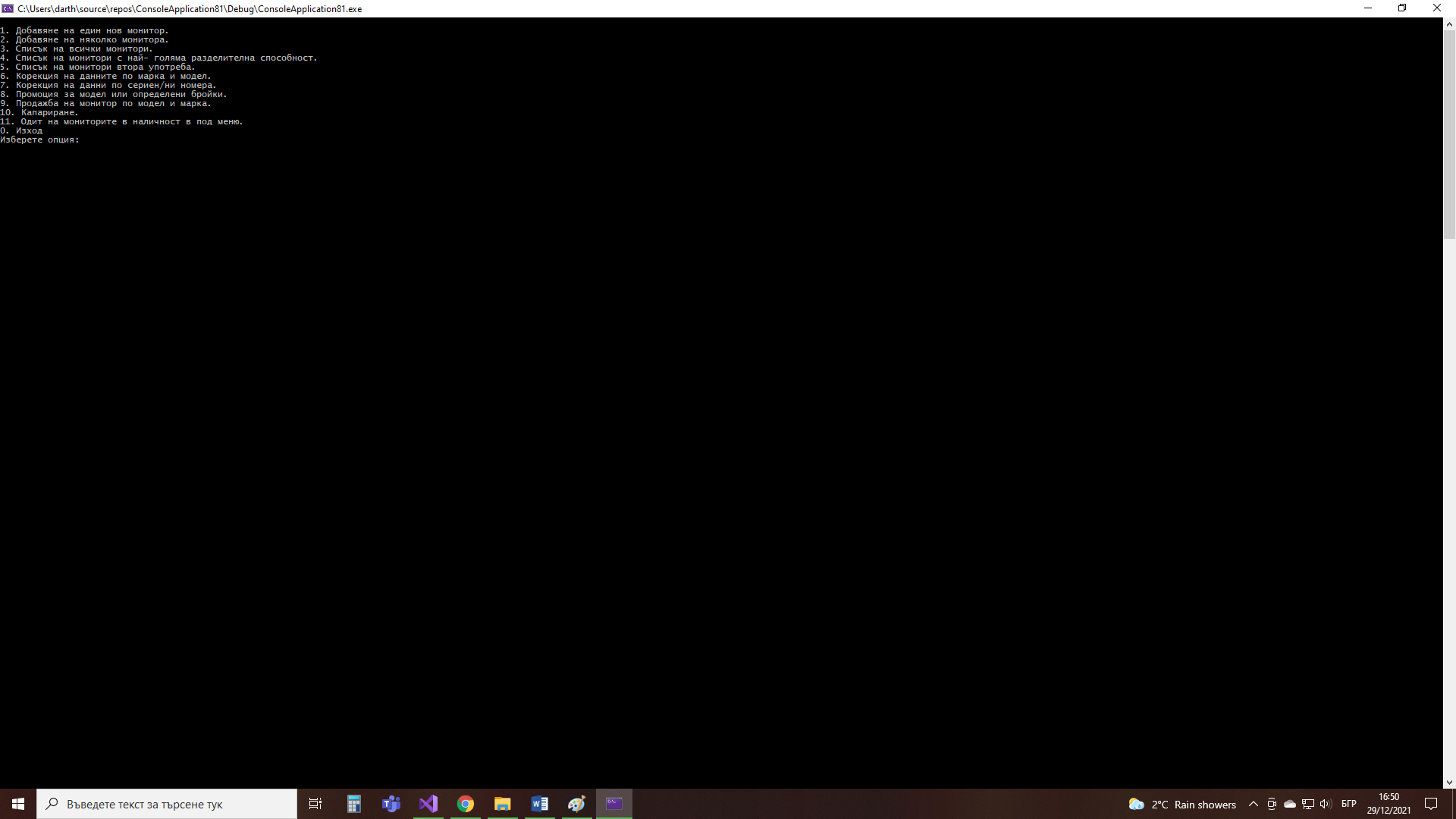
Очакван резултат от изпълнението на конкретния фрагмент от проекта

Ако отговорът е „не“, програмата приключва, ако отговорът е „да“, данните на монитора се записват в отделен текстов файл.

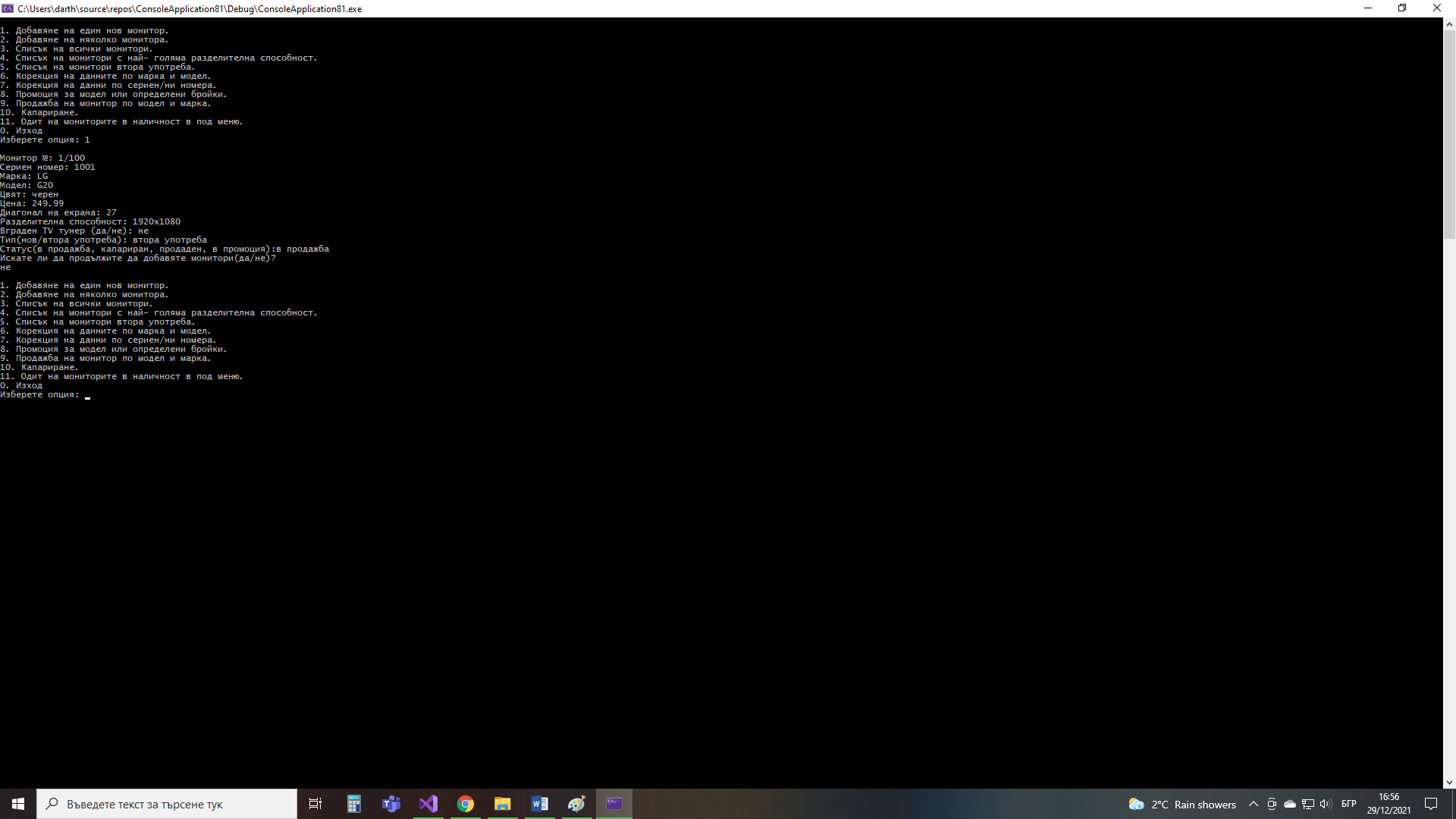
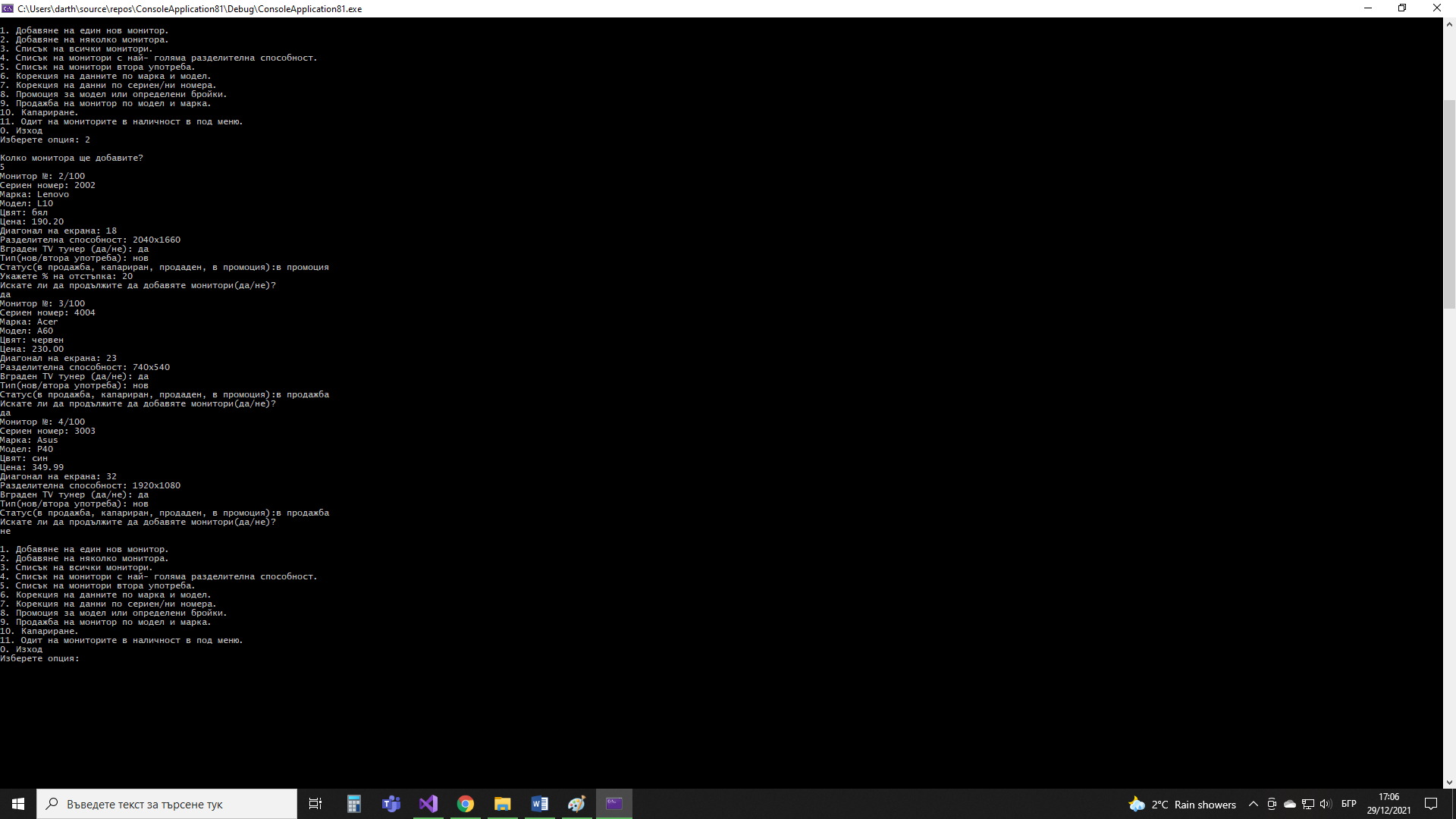
Примерно действие на програмата

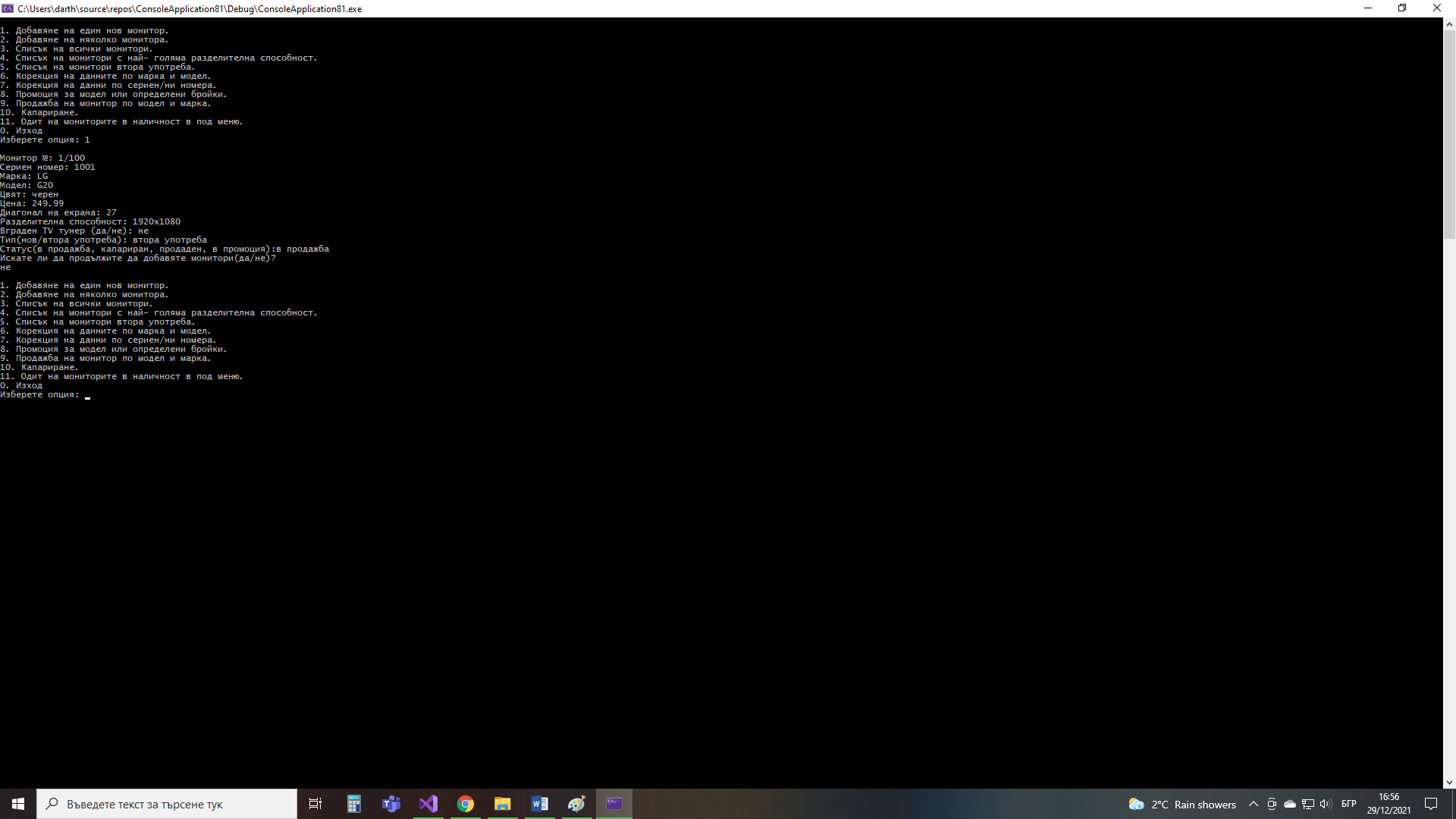
Условие A

Снимка на изгледа с примерни входни данни

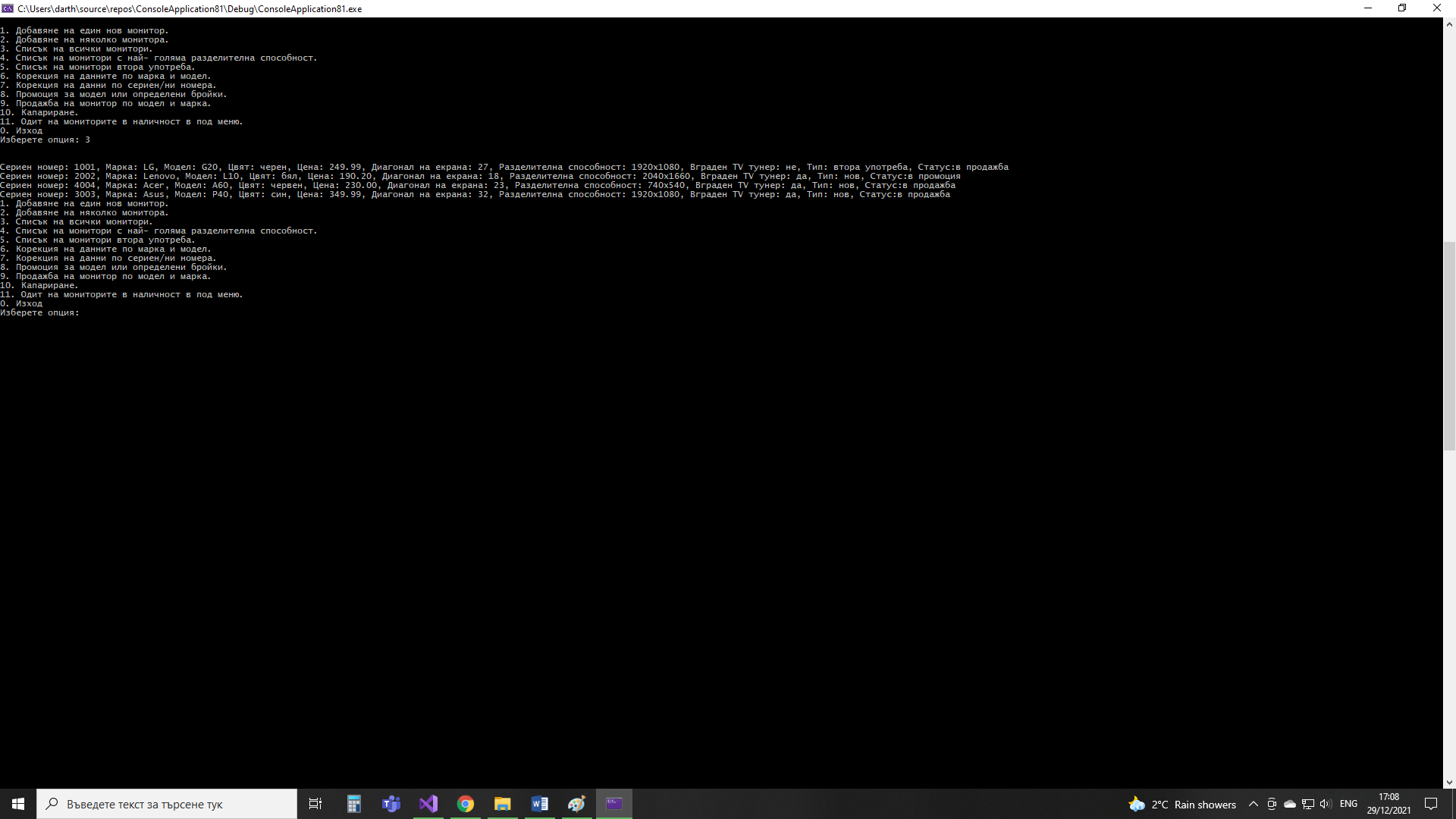
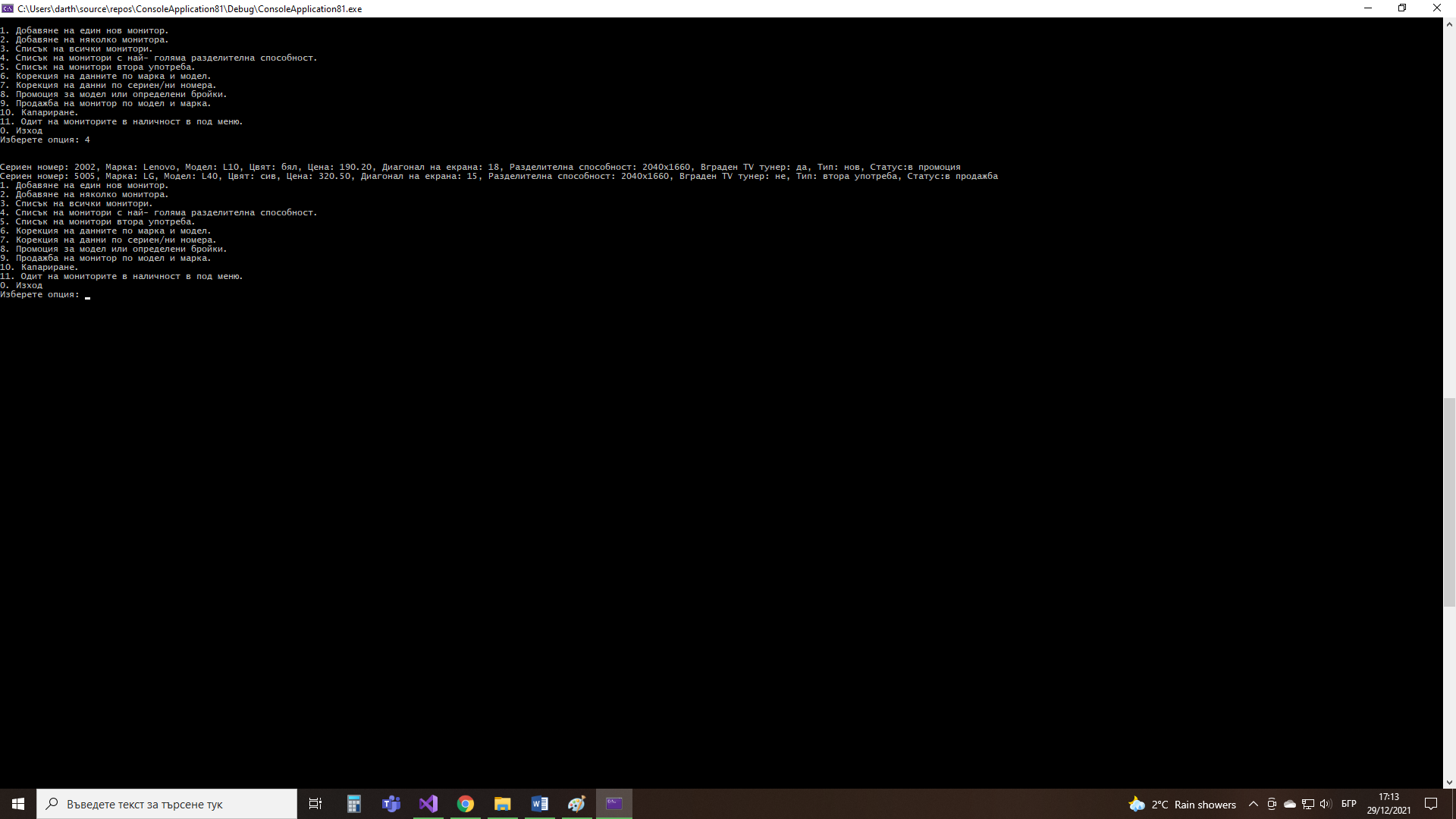
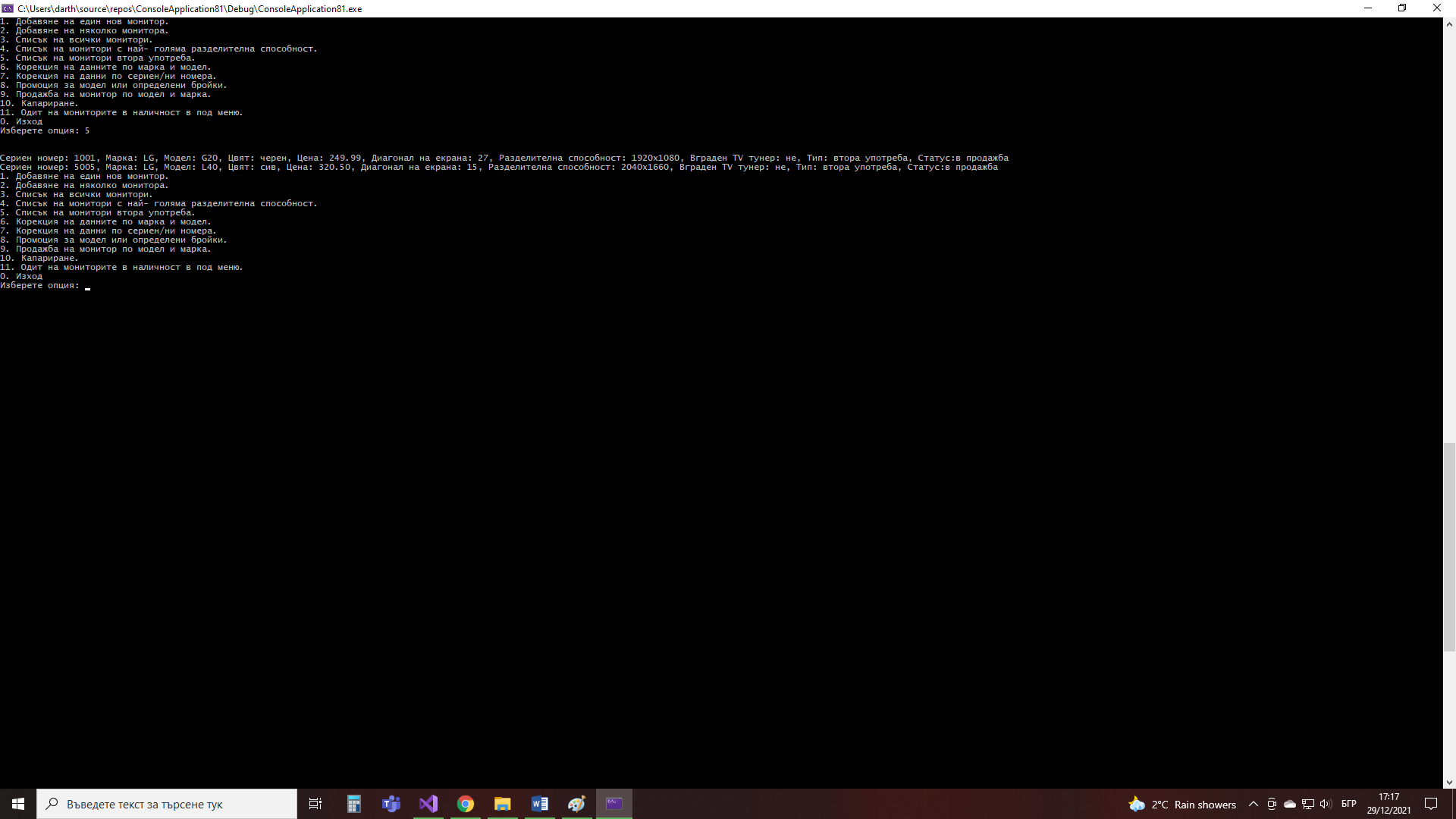
Снимка на изгледа с примерни изходни данни  


Условие B

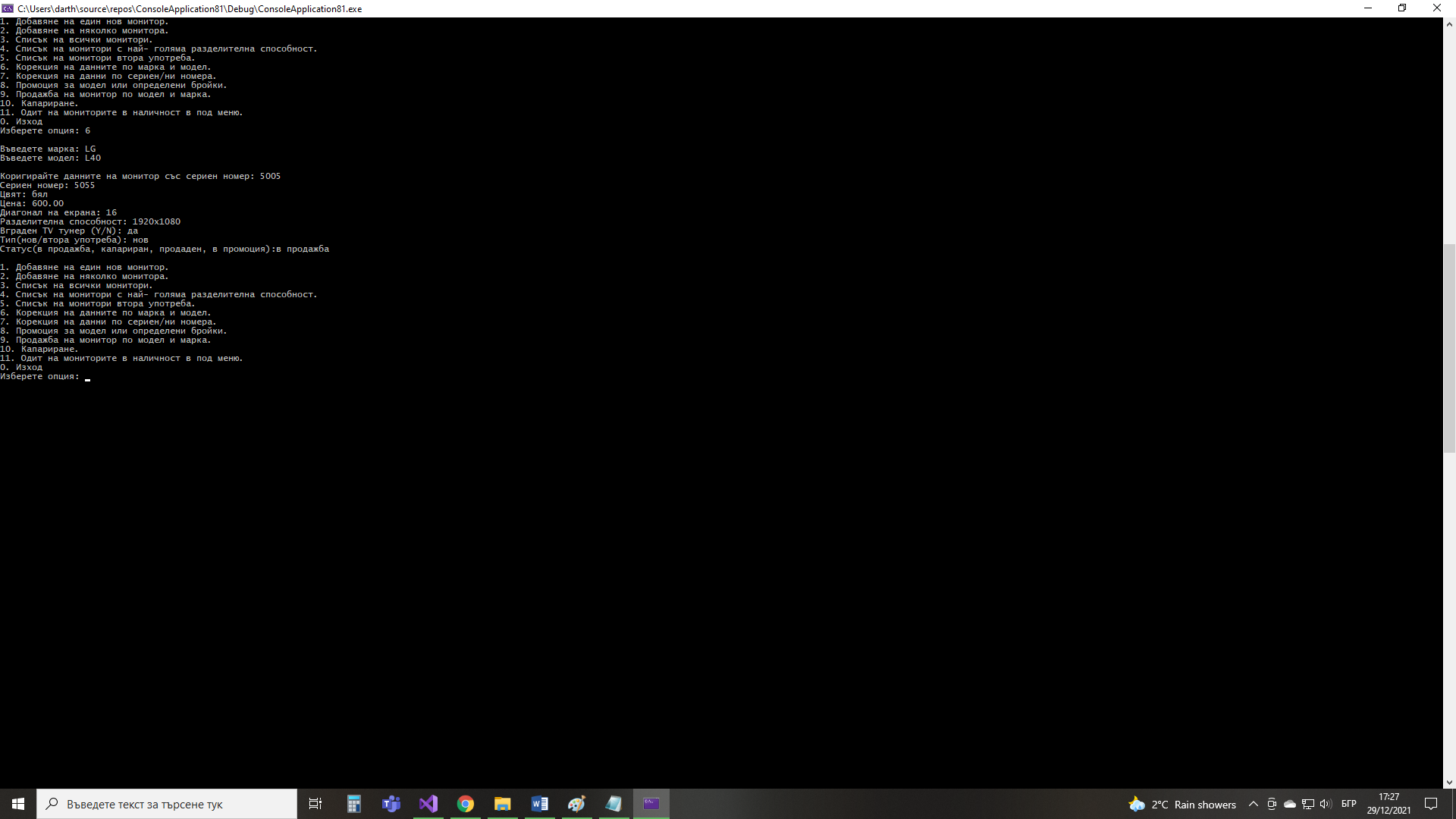
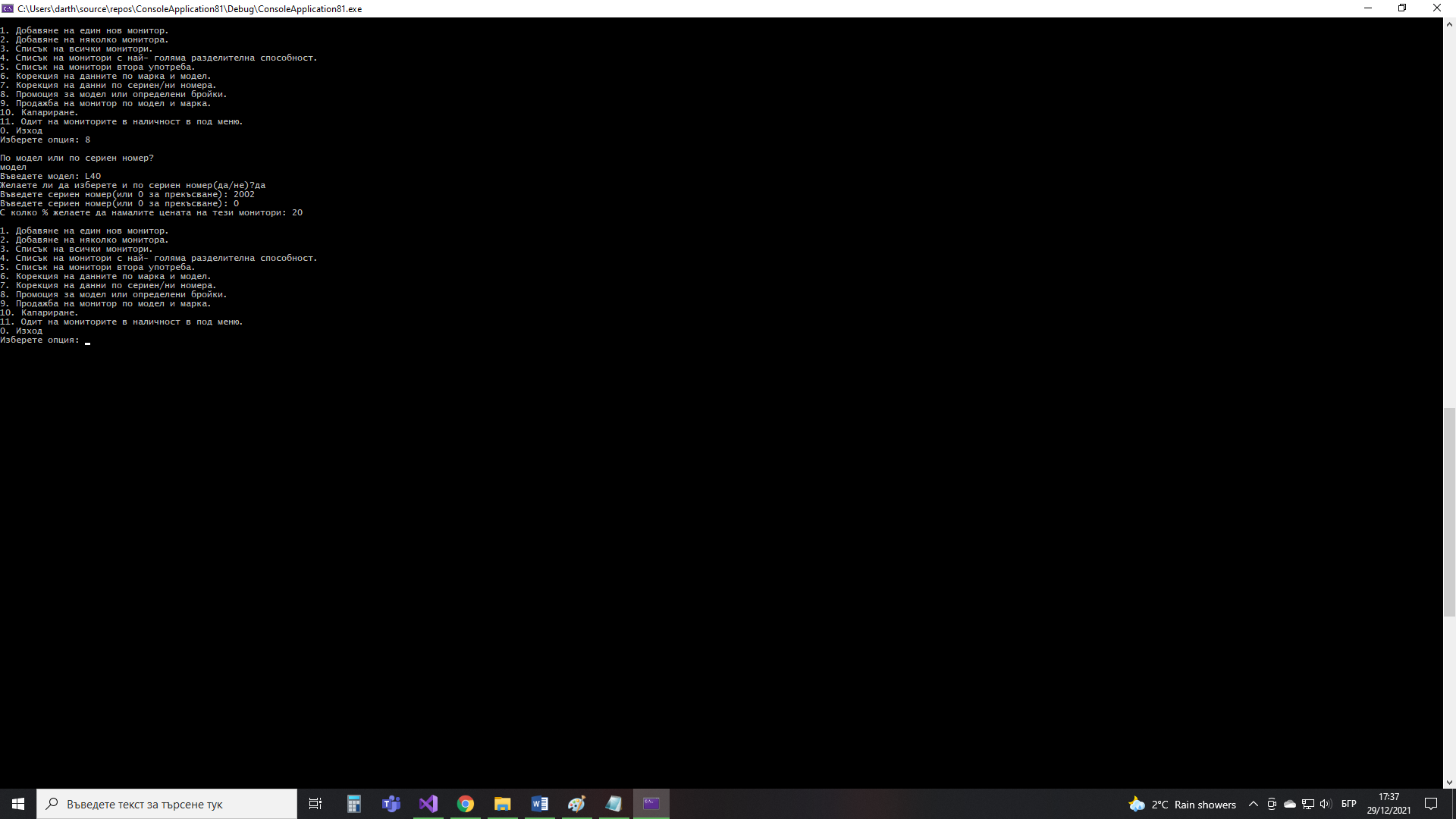
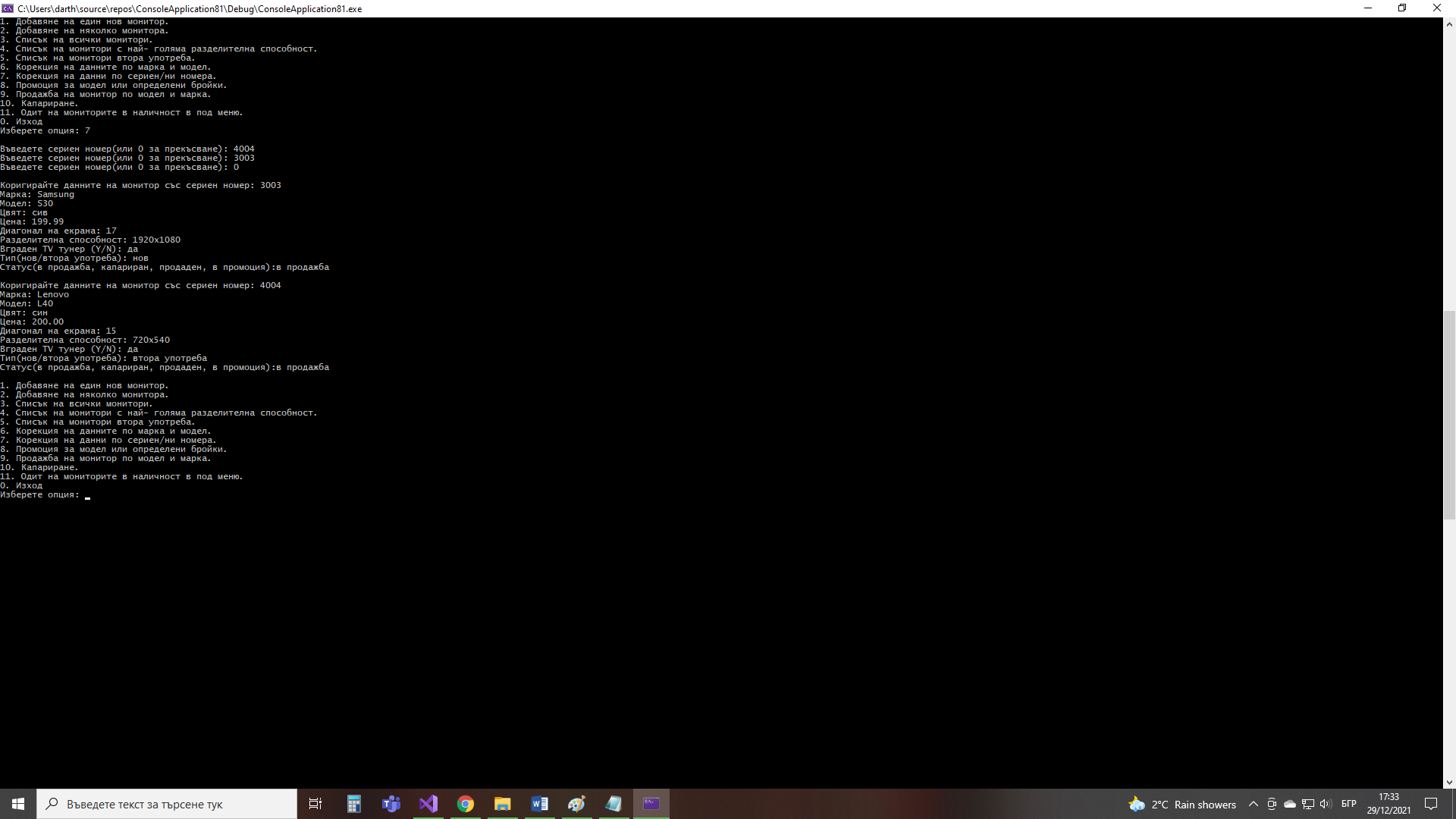
Снимка на изгледа с примерни входни данни   
a.  
  
  
  
b.  


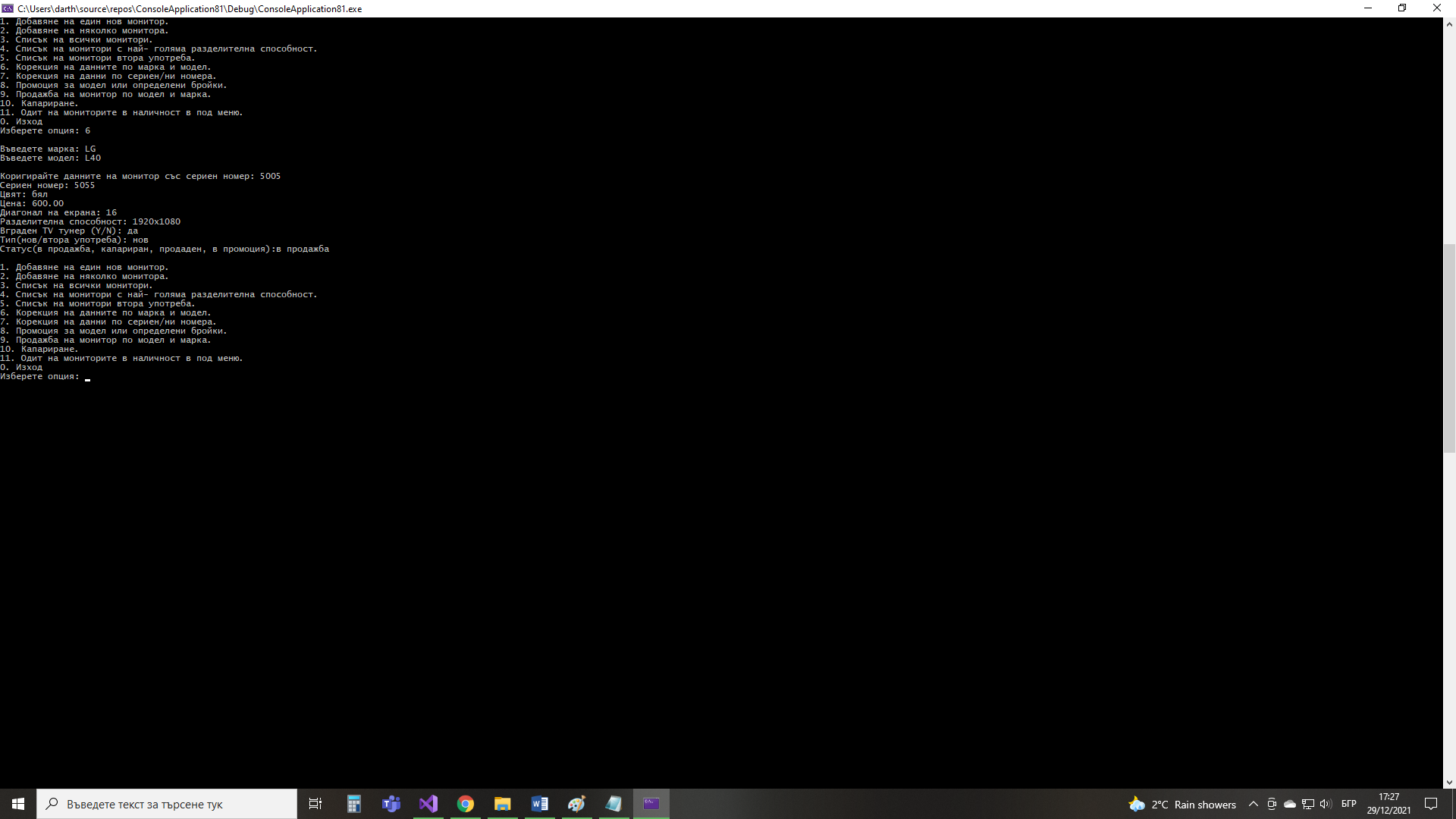
Снимка на изгледа с примерни изходни данни 

Условие C

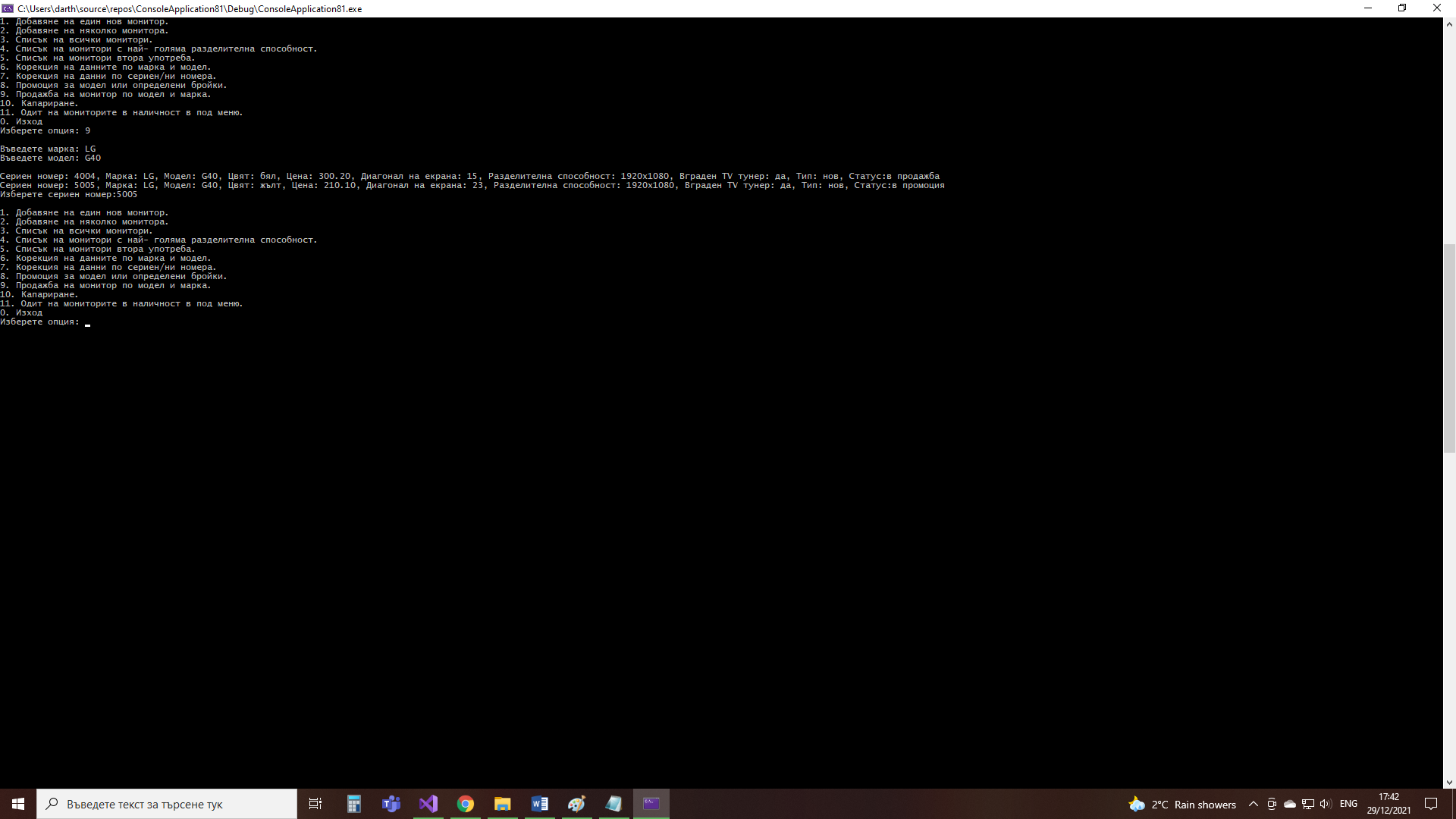
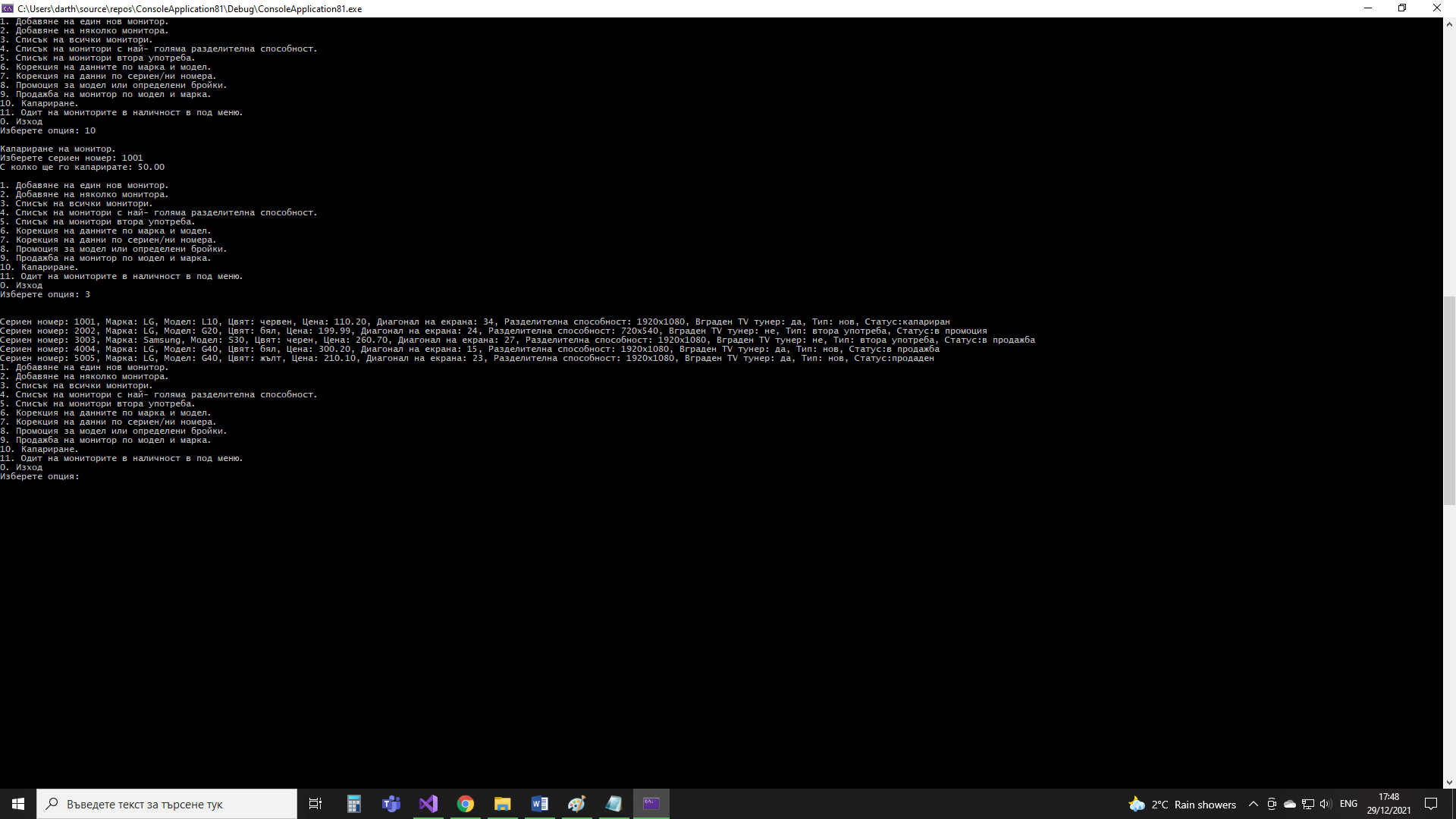
Снимка на изгледа с примерни изходни данни  
a.   
  
b.  
  
c.

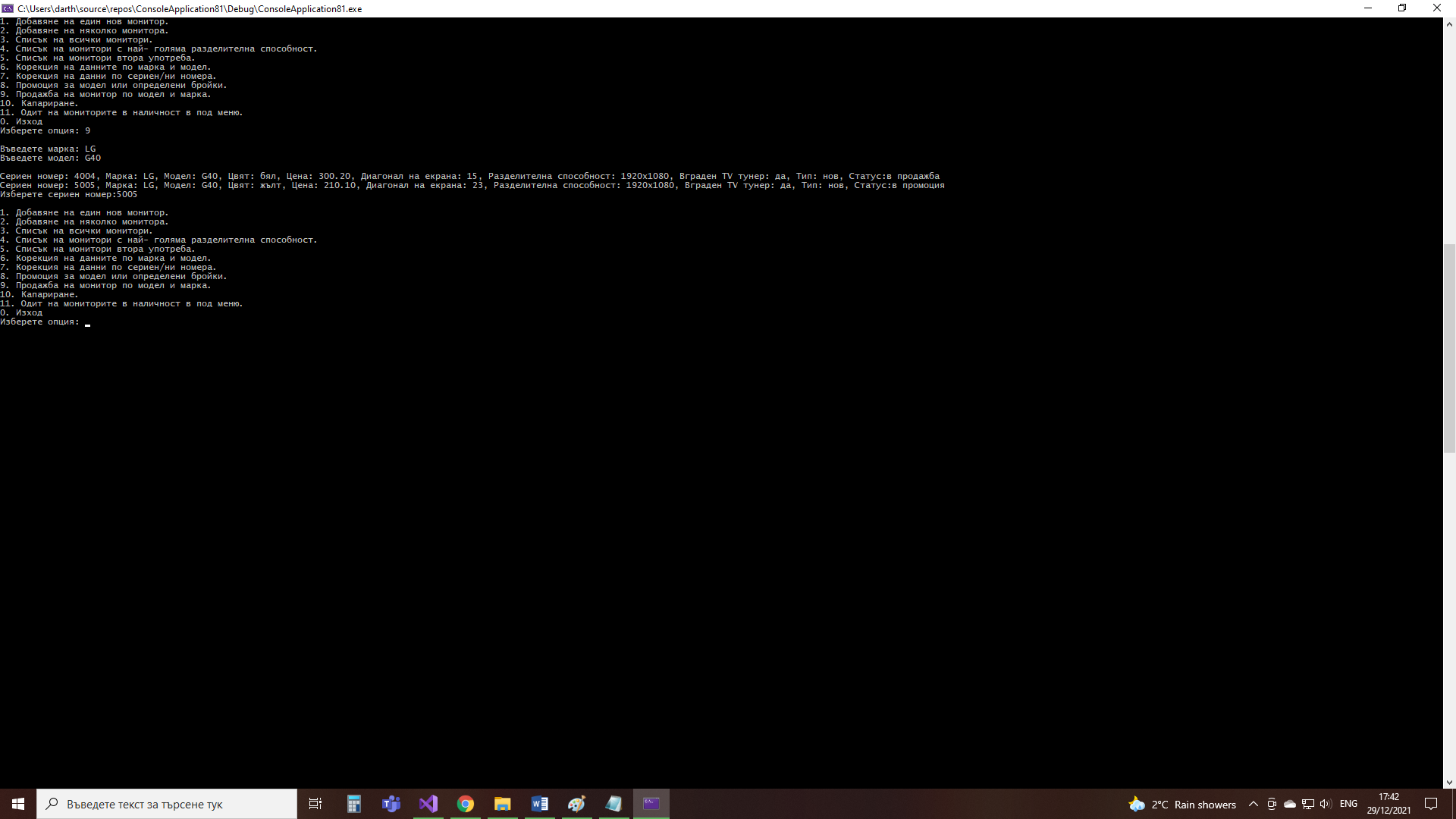
Условие D

Снимка на изгледа с примерни входни данни  
а.   
  
  
  
  
  
  
  
  
  
b.  
  
  
c.

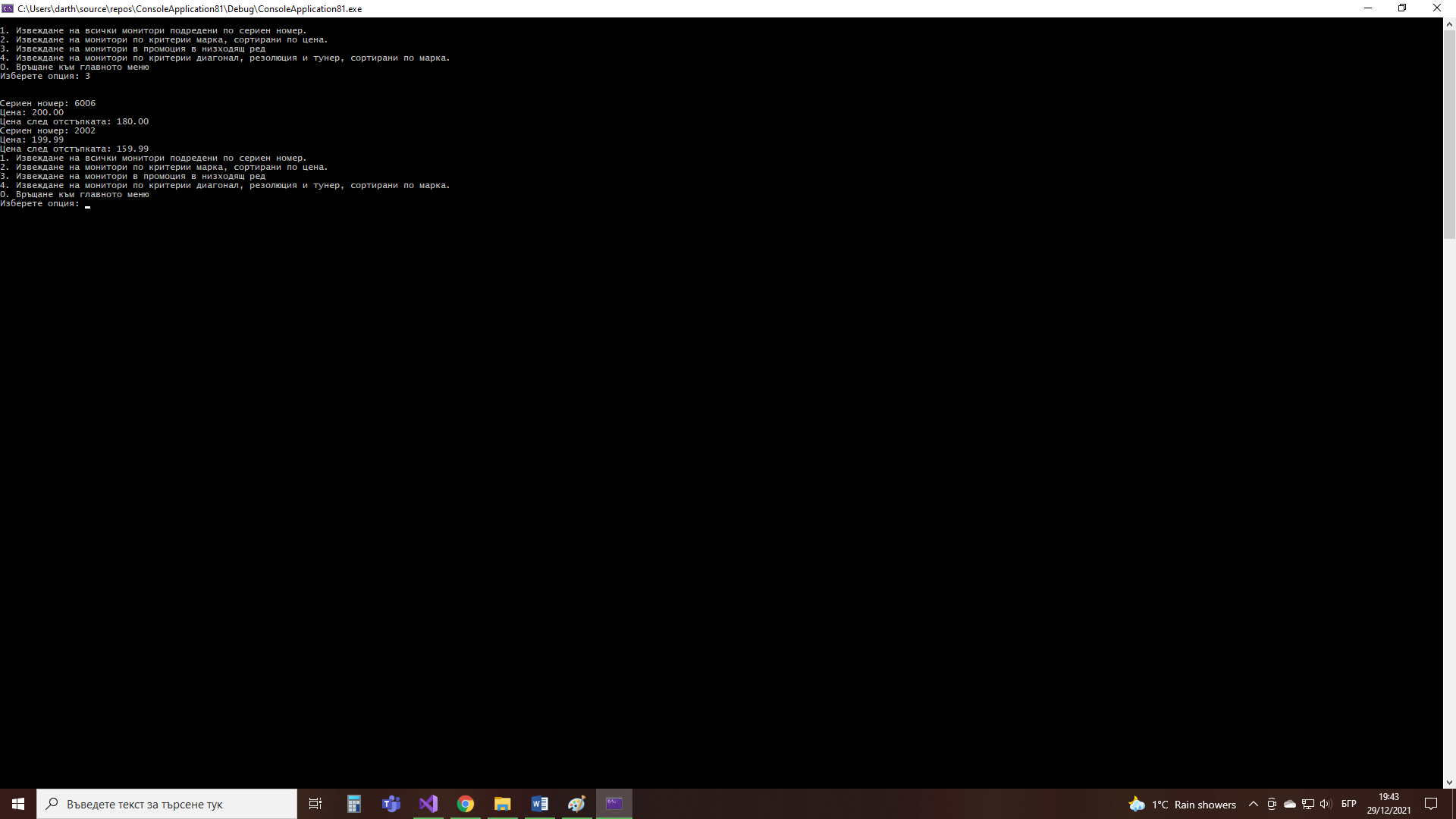
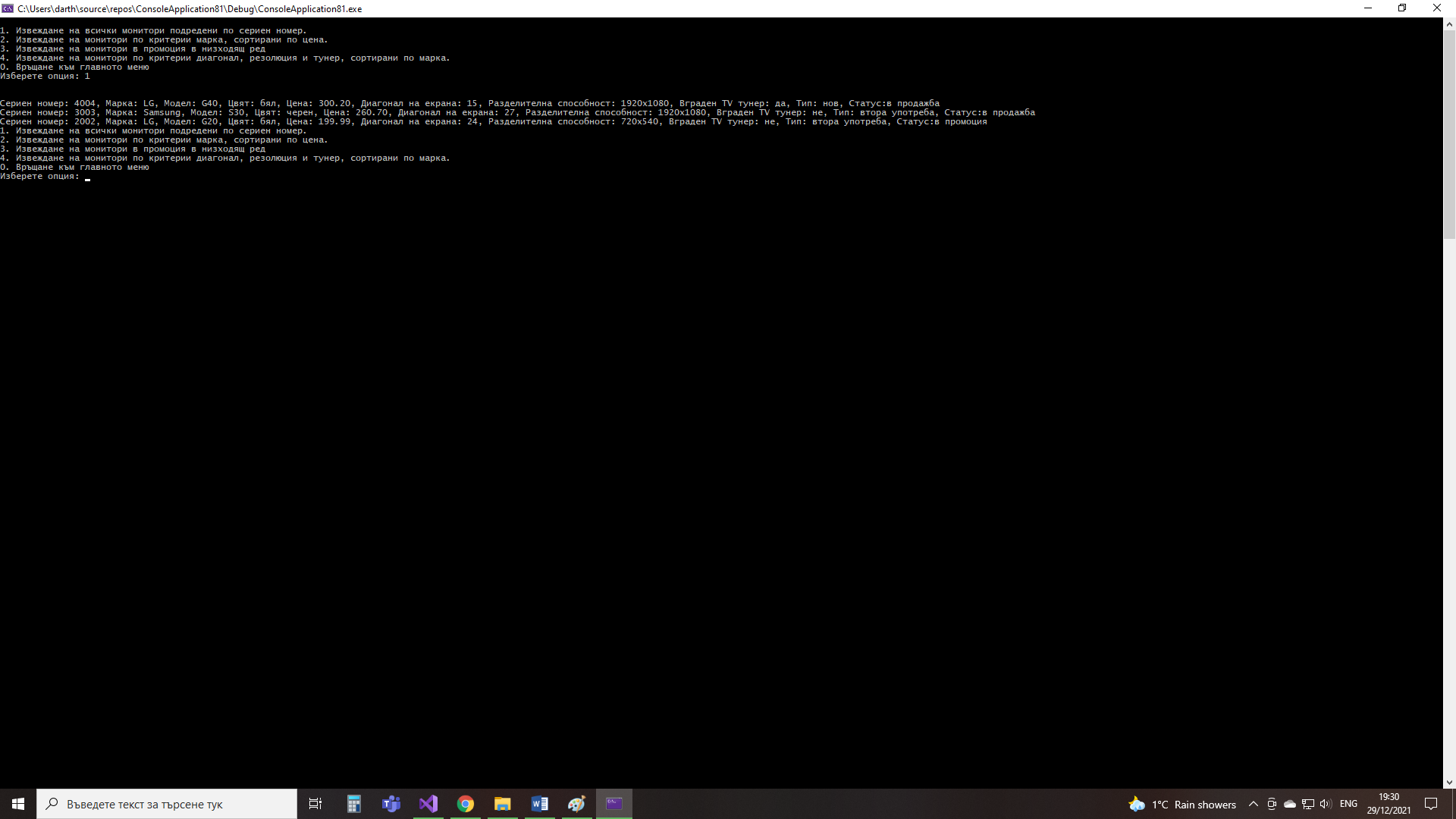
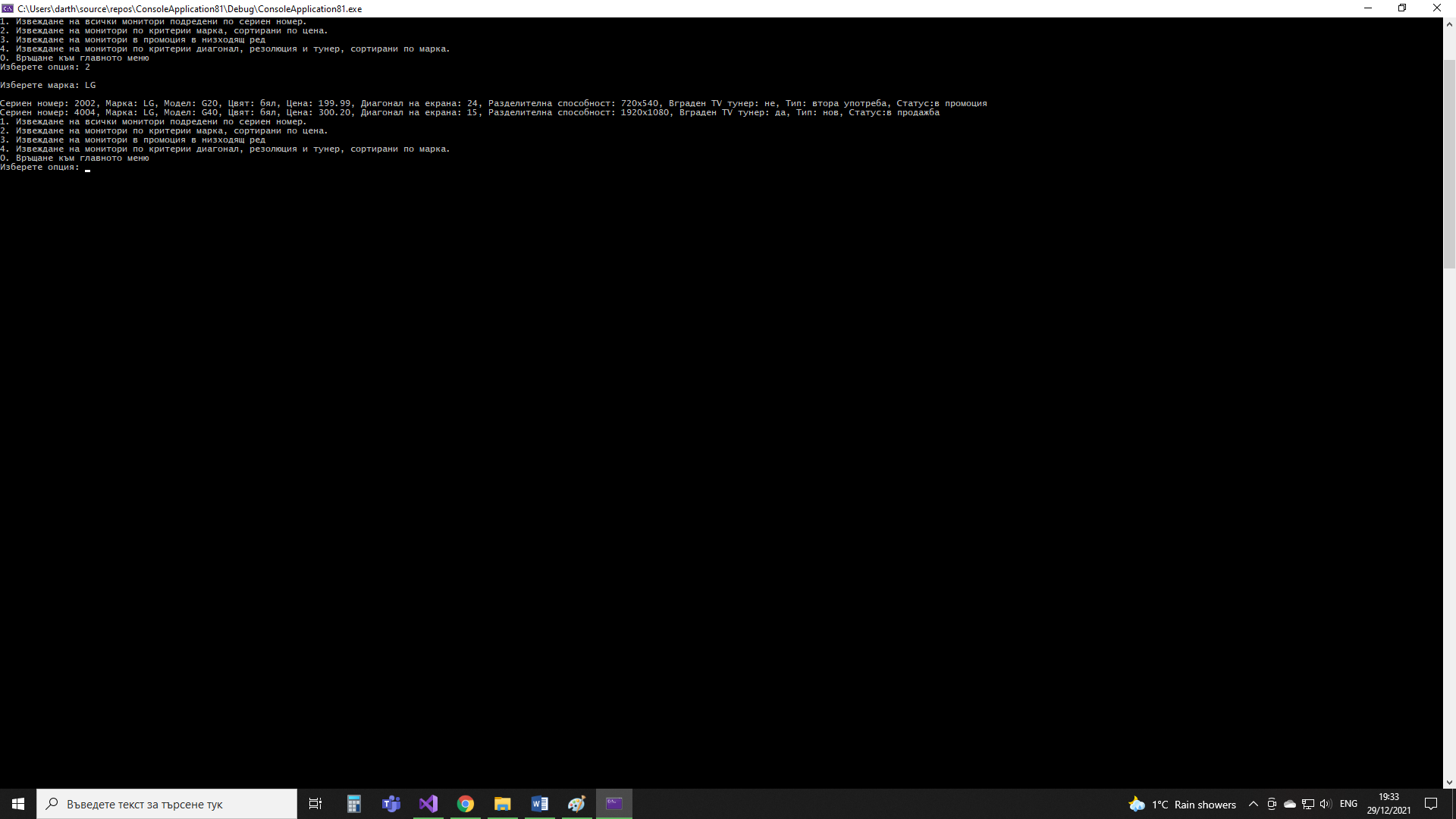
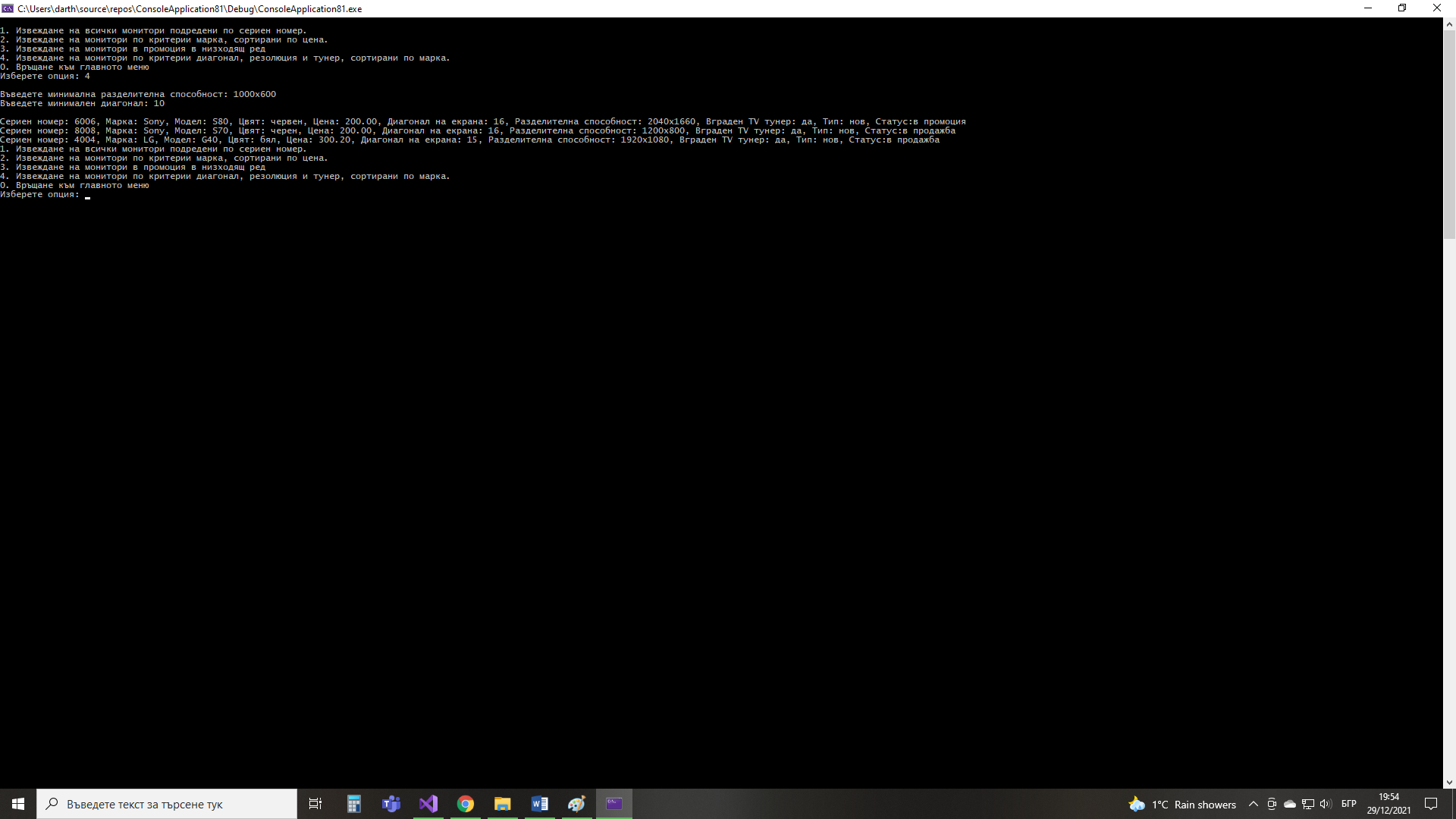
Снимка на изгледа с примерни изходни данни

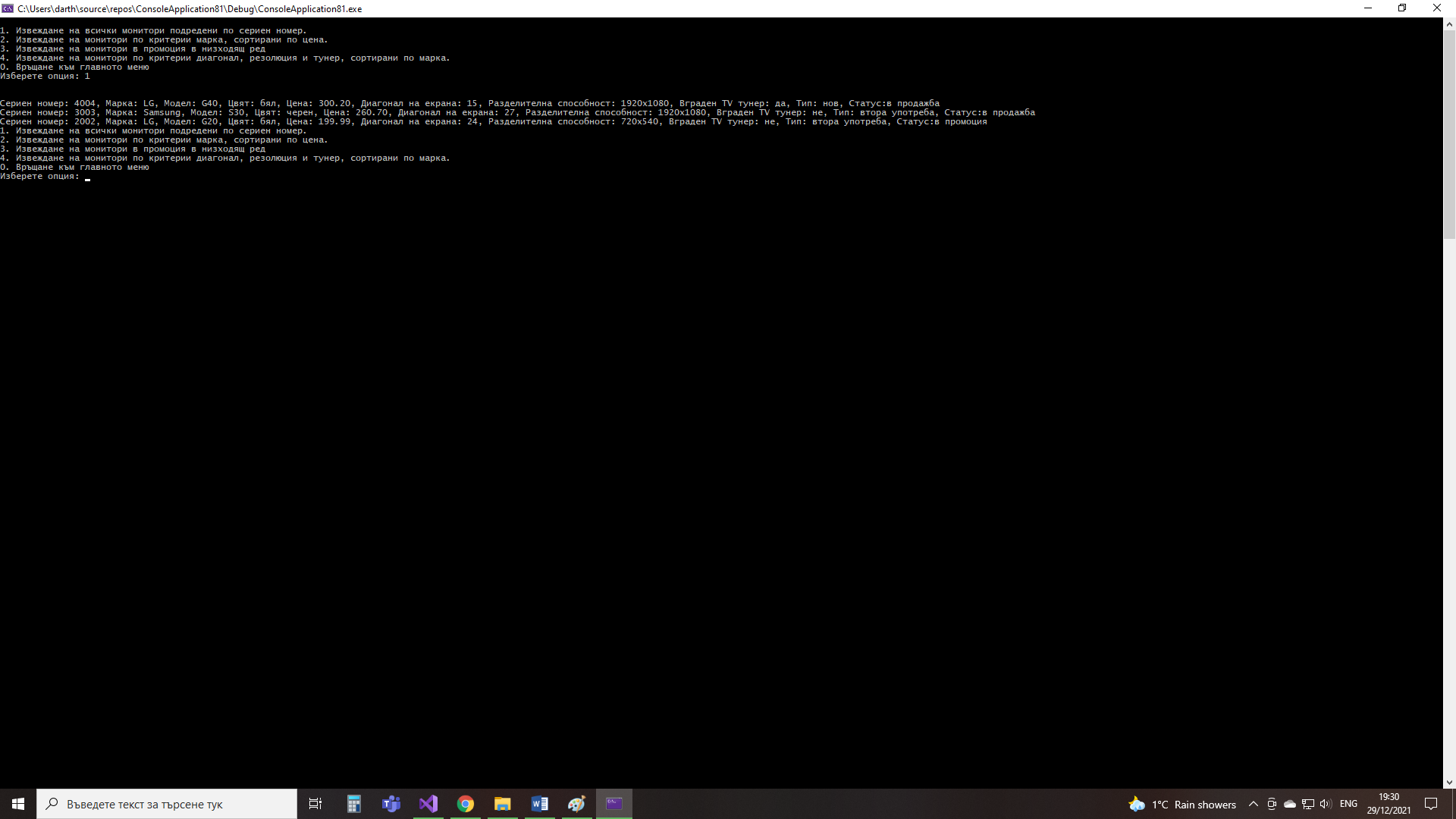
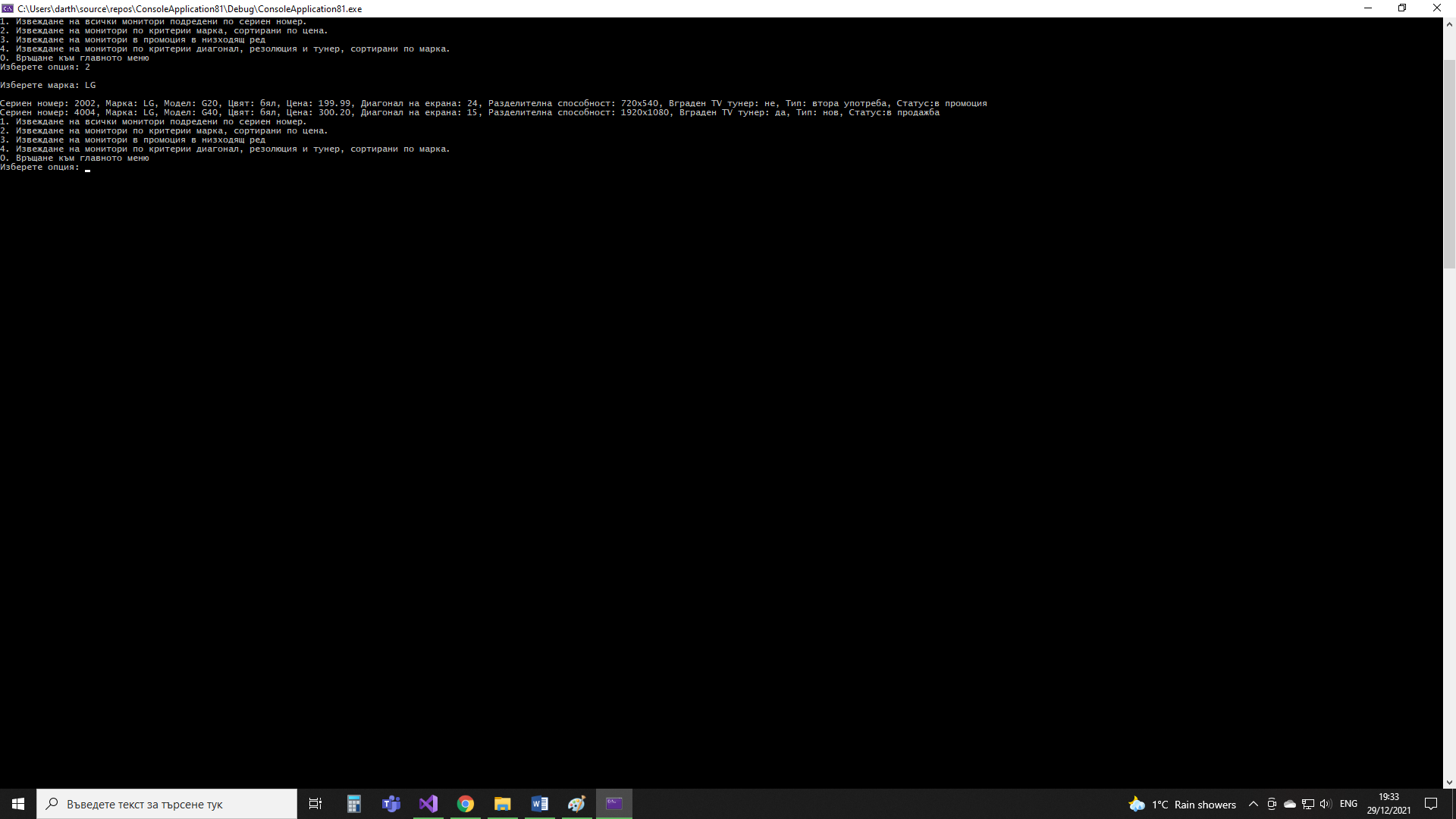
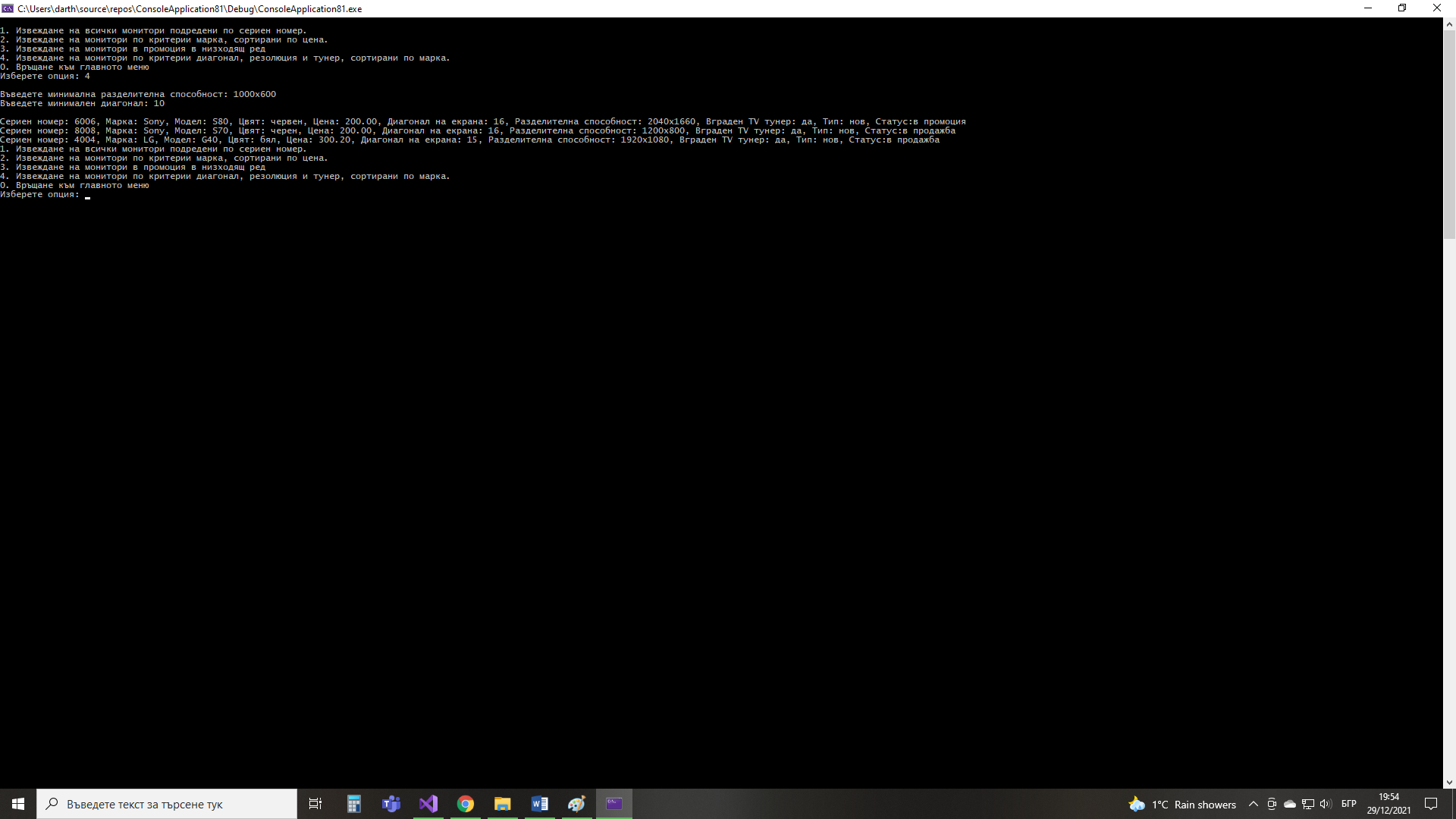
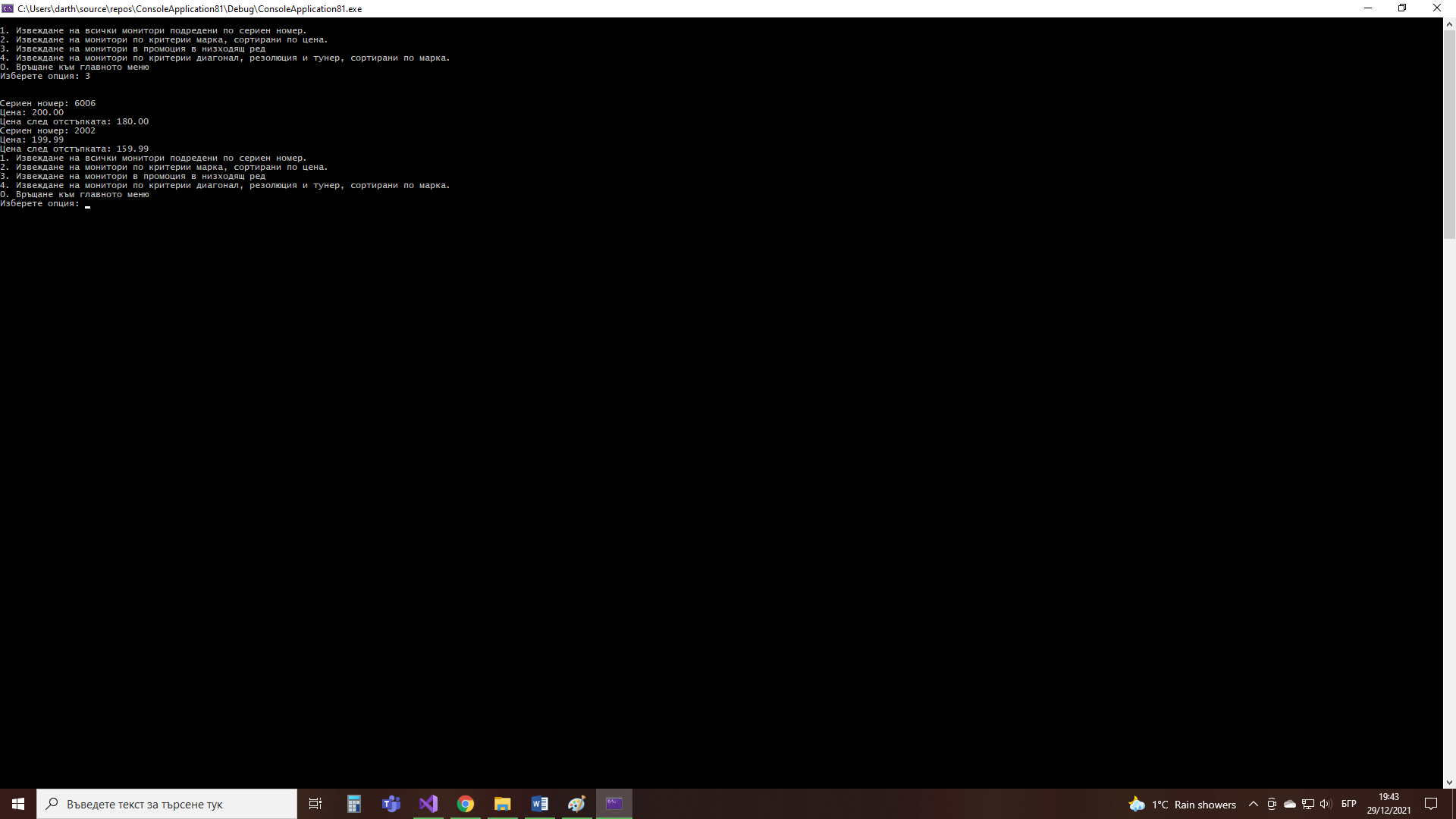
Условие E

Снимка на изгледа с примерни входни данни  
а.   
b.  
 

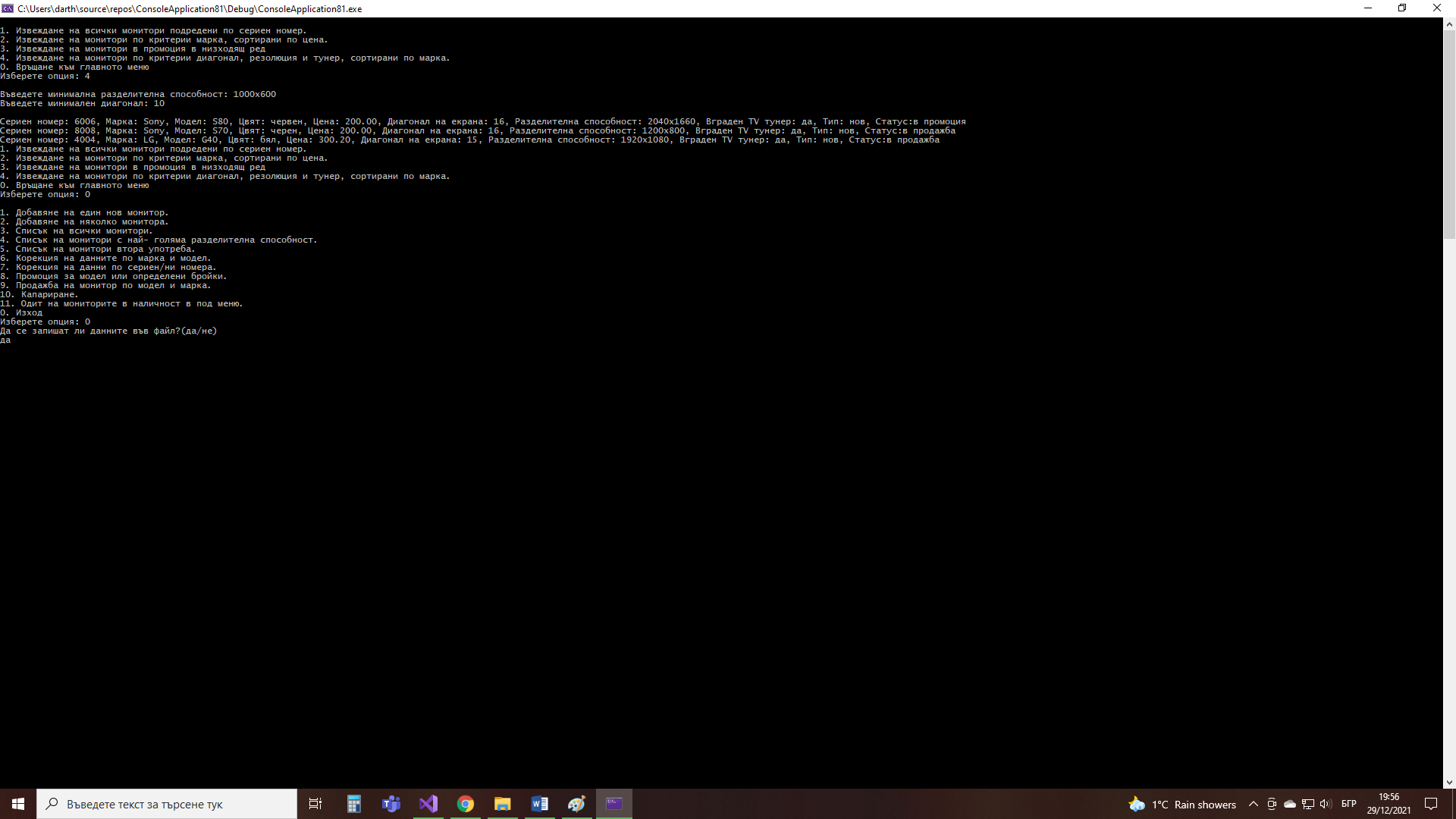
Снимка на изгледа с примерни изходни данни  
 

Условие F

Снимка на изгледа с примерни входни данни  
a.  
   
b.  
   
c.   
  
  
  
  
d.  
 

Снимка на изгледа с примерни изходни данни  
a.   
b.  
c.  
   
d.

Условие G

Снимка на изгледа с примерни входни данни

Снимка на изгледа с примерни изходни данни