****

**INF212**

**ALGORITHMS AND PROGRAMMING II**

**PROJECT-1**

**Sale Program**

**Deadline is April 1, 2021 at 23:00.**

**Projects that are not delivered on time are not accepted.**

**Upload the project to the Project 1 assignment section of the INF212 class.**

**The questions can be asked to course lecturer Dr. Tuba GÖZEL and teaching assistant Süleyman TUNCEL and Mehmet Türker TAKCI.**

**Complete your project, fill the report by using attached word file   
(INF212\_Project\_1\_Report\_Template.docx).**

**Upload the project codes and the report.**

**You can compress the project codes if number of the files is more than ten.**

**The file name of report should be as *P1\_ID\_FirstLetterofName\_.docx*   
(i.e. *P1\_141024001\_YKZ.docx* ).**

****

**INF 212 Algorithms and Programming II**

**2020-2021 Spring**

**Electronic Engineering**

**Proje ID# Project Name**

**\_Program Name\_**

|  |  |  |
| --- | --- | --- |
| **School ID** | **Name** | **Surname** |
| **1901022022** | **Selimhan** | **Aygün** |
|  |  |  |
|  |  |  |

|  |
| --- |
| PROJECT OBJECTIVE  Briefly describe the objectives of your project |
| Program kısaca bir satış otmasyonu. Programı kullanacak kişi ise bir satış elemanı olabilir.  Bu program sayesinde satış elemanı içerisinde listelenen ürünleri içerisinde listelenen kişilere satabilir, bunları görüntüleyebilir, ürüne ve müşteriye ait bilgilerin analizini yapabilir. |
| PROBLEM  Briefly describe the problem you given to solve |
| Amacım bir satış otomasyonu yazmak olduğu için problemde elemanları listeleme, listeye ekleme ve listeden yazdırma bulunmaktadır. Ayrıca bu listelere ait bilgileri analiz etme başka bir değişle kullanma problem dahilindedir. |
| ANALYSIS  Analyze of the problem |
| Problemi çözmek için listeleme araçlarını düzgün bir şekilde kullanmak gerekir. Ben çözüm için bağlı liste kullandım ve işlemlerimi bağlı listeler üzerinden gerçekleştirdim. Analiz kısmında ise elemanları tutması ve kıyas etmesi için dizilerden faydalandım. Dizilerdeki elemanları karşılaştırıp düzgün sonuçlar elde edebildim. |
| DATA REQUIREMENTS  Specify all data requirements of the problem |
| **Structured Data Type** |
| struct customer {  int ID;  char name[50];  unsigned int type: 1;  double x\_coord;  double y\_coord;  struct customer \*next;  };  typedef struct customer CustomerNode;  typedef CustomerNode \*CustomerPtr;  struct product {  int ID;  char name[50];  unsigned int type : 4;  double price;  struct product \*next;  };  typedef struct product ProductNode;  typedef ProductNode \*ProductPtr;  struct purchased {  int ID;  int invoice\_ID;  int customer\_ID;  int product\_ID;  double cost;  int quantity;  double s\_fee;  struct Purchased \*next;  }; |
| **Problem Constants** |
| int choice,cho, invoice\_id=1, customer\_id,product\_id,quantity, purchased\_id=0, j;  int cho1, cho2, cho3, cho4, cho5, cho6, cho7,cho8, cho9;  unsigned int customer\_type,product\_type;  double cost;  double price, s\_fee; |
| **Problem Inputs** |
| int cho1, cho2, cho3, cho4, cho5, cho6, cho7,cho8, cho9;  int customer\_id;  int product\_id; |
| **Problem Output** |
| void print\_AllCustomers(CustomerNode \*r);  void CustomerTypeList(CustomerNode \*r, unsigned int type);  void print\_SingleCustomer(CustomerNode \*r, int customer\_id);  void print\_AllProducts(ProductNode \*r);  void ProductTypeList(ProductNode \*r, unsigned int type);  void print\_SingleProduct(ProductNode \*r, int product\_id);  void ListCustomerPurchased(PurchasedNode \*r, ProductNode \*z, int customer\_id);  void ListACustomerTotal(PurchasedNode \*r, int customer\_id);  void AllCustomersTotal(PurchasedNode \*r);  void TotalShippingFee(PurchasedNode \*r);  void TotalPurchaseQuantity(PurchasedNode \*r, int product\_id);  void TotalPurchaseType(ProductNode \*r,PurchasedNode \*z, unsigned int product\_type);  void print\_AllCosts(PurchasedNode \*r); |
| DESIGN  ALGORITHM  Design algorithm of the problem |
|  |
| HEADER F;ILE(S) |
| #ifndef PRODUCT\_\_H\_  #define PRODUCT\_\_H\_  struct customer {  int ID;  char name[50];  unsigned int type: 1;  double x\_coord;  double y\_coord;  struct customer \*next;  };  typedef struct customer CustomerNode;  typedef CustomerNode \*CustomerPtr;  struct product {  int ID;  char name[50];  unsigned int type : 4;  double price;  struct product \*next;  };  typedef struct product ProductNode;  typedef ProductNode \*ProductPtr;  struct purchased {  int ID;  int invoice\_ID;  int customer\_ID;  int product\_ID;  double cost;  int quantity;  double s\_fee;  struct Purchased \*next;  };  typedef struct purchased PurchasedNode;  typedef PurchasedNode \*PurchasedPtr;  void menu(void);  void set\_CustomerInfo(CustomerNode \*r, CustomerNode \*customer\_info);  void set\_ProductInfo(ProductNode \*r, ProductNode \*product\_info);  int CheckCustomerID(CustomerNode \*r, int customer\_id);  int CheckProductID(ProductNode \*r, int product\_id);  double CheckShippingFee(CustomerNode \*r, int customer\_id);  double CalculateCost(ProductNode \*r, int product\_id, int quantity);  void PurchasedStorage(PurchasedNode \*r, int purchased\_id, int invoice\_id, int customer\_id, int product\_id, double cost,int quantity, double s\_fee);  void print\_AllCustomers(CustomerNode \*r);  void CustomerTypeList(CustomerNode \*r, unsigned int type);  void print\_SingleCustomer(CustomerNode \*r, int customer\_id);  void print\_AllProducts(ProductNode \*r);  void ProductTypeList(ProductNode \*r, unsigned int type);  void print\_SingleProduct(ProductNode \*r, int product\_id);  void ListCustomerPurchased(PurchasedNode \*r, ProductNode \*z, int customer\_id);  void ListACustomerTotal(PurchasedNode \*r, int customer\_id);  void AllCustomersTotal(PurchasedNode \*r);  void TotalShippingFee(PurchasedNode \*r);  void TotalPurchaseQuantity(PurchasedNode \*r, int product\_id);  void TotalPurchaseType(ProductNode \*r,PurchasedNode \*z, unsigned int product\_type);  void print\_AllCosts(PurchasedNode \*r);  #endif |
| IMPLEMENTATION  Explain the method applied for the problem solution |
| En başta menu() adında bir fonksiyon ile menü hazırladım. Ve switch-case kullanarak menüdeki seçeneklerin seçilebileceği bir ara yüz tasarladım.  Ardından struct’taki yapıları array kullanarak hazır bir şekilde yerleştirdikten sonra bunların hepsini bir set fonksiyonu ile bağlı listelere atadım. Inputlarla aldığım id’leri test edip yanlış girdilere dair çıktılar hazırladım. Listeye eklenecek şeyleri pointer ve nodeları kullanarak bağlı listeye atadım. Yazdırılması gerektiğinde de while ile nodelar üzerinden ilerleyerek gerekli şartlar altında yazdırılmasını ayarladım.  Problemdeki en zor problemlerden birisi olan en sık tekrar eden elemanı bulmayı ise array kullanarak yaptım. Gelen ürün numarasını dizi içinde aynı sayıya karşılık gelecek bir elemana atadım ve ürün numarasından kaç tane olduğunu gösteren quantity’i dizinin o sayıdaki karşılığına atadım. Ardından maksimum 100 sayı olacak şekilde bu diziyi ilerletip en fazla geçen sayıyı buldum.    Diğer bir büyük problem olan faturadaki en yüksek miktarı bulmayı ise yine dizi kullanarak yapmaya çalıştım. Bağlı listeledeki elemanları Dizi içindekileri kontrol ettirip en yüksek olanı maksimum adlı değişkene atadım. En düşüğü ise minimum adlı değişkene atadım ancak çıktı kısmında beklemediğim hatalarla hatalarla karşılaştım. |
| TESTING  Briefly describe how you test your code |
| Bağlı liste kısımlarındaki sorunları kodu çalıştırıp aralara printf’lerle fonksiyondaki değerleri yazarak değer atamasının gerçekleşip gerçekleşmediğini ve hatanın nerede olduğunu tespit ettim.  Ayrıca ürünü düzgün bir şekilde kullanmayan bir kullanıcı gibi girişler yapıp muhtemel hata ve eksiklikleri tespit ettim. |
| USER’S GUIDE  (Describe briefly how the user can use your program (input parameters of program, output of program, possible limitations, possible errors, etc.) |
| Kullanıcı Menünün 2. Ve 3. Seçeneklerinde yer alan verilerle ilgili kısımları programı açar açmaz rahat bir şekilde kullanabilir, herhangi bir girdiye bu kısımda gerek yoktur. Sale kısmında ise programın yönergelerine göre girişleri yaptıktan sonra programın geri kalanına erişim sağlayabilir.  Menünün 4. Seçeneğinin 2. Kısmda yer alan bir müşterinin satın aldığı toplam tutarı düzgün bir şekilde gösteriyor ancak maksimum ve minimum göstergeleri çalışmıyor. Birden fazla ürün eklenmesi durumunda program yine satın alınan tutarı gösteriyor ancak bir kaç saniye sonra hata verip sonlanıyor.  Menünün (4.3) Tüm müşterinin satın aldığı ürünlerin toplam tutarı ve (4.4) Müşterilerin kargo ücreti hesaplamalarında istenmeyen belirsiz bir çıktı olduğu için doğru olan kısmı bozmaktadır.  Menünün 5.3 Tüm ürünlerin toplam satış tutarı kısmında ise yine aynı şekilde istenmeyen bir çıktı tarafından toplam tutar bozulmaktadır. |
| GENERAL FLOWCHART OF THE PROJECT |
|  |
| CONCLUSION AND REMARKS  You can or can(not) put any remarks on your work. For example;  is your program works or not, if not why  which troubles during performance of project you faced with and how you overcome them;  is your program user friendly and how you can improve it;  what you achieved during performance of project; and so |
| Programı yazarken ve test ederken bir çok zorlukla karşılaştım. Öncelikle Bağlı listelere hiç hakim değilken üzerinde çokça durup mantığını ve işleyişini anladım. Bu benim açımdan çok faydalı oldu. Ayrıca çok emin olduğum bir kodun bile bir yerde patlaması sabrımın sınırlarını zorladı ve bana büyük ölçüde tahammül seviyesi kazandırdı.  Programım gayet sade ancak clear komutunu hiç kullanmadığım için çıktılar bazen iç içe girebiliyor ve konsoldak görünümünde sapmalar olabiliyor. Bunların biraz daha üstünde durabilirdim.  Programım %85 oranında düzgün çalışıyor. Çalışmayan kısımlardaki hataları tespit etmeye çalıştım ancak kodun çalışan kısımlardan bir farkı olmadığı için başarısız oldum. Muhtemelen datalarla ilgili işlemler yaptığımız için herhangi küçük bir hata büyük bir hataya dönüşüyor ve programın o kısmını bloke ediyor. |
| REFERENCES  Put the list of references and sources (books, links to websites, videos, etc), which you used for project. |
| <https://www.log2base2.com/data-structures/linked-list/inserting-a-node-at-the-end-of-a-linked-list.html>  <https://www.log2base2.com/data-structures/linked-list/searching-a-node-in-singly-linked-list.html>  https://www.codeproject.com/Questions/809285/expected-expression-before-token  https://stackoverflow.com/questions/22398987/most-frequent-element-in-a-sequence-using-arrays-in-c |

|  |
| --- |
| APPENDIX |
| PROJECT CODES  Put here code of your project. |
|  |