

RELAZIONE ELABORATO
ARCHITETTURA DEGLI ELABORATORI
ANNO 2022/2023
(ASSEMBLY)

Mattevi Mirco VR486372

SPECIFICHE PROGETTO

Realizzare un programma Assembly per la gestione del menù cruscotto di un'automobile. Il menù dovrà visualizzare e permettere l'impostazione delle seguenti informazioni: data, ora, impostazione blocco automatico porte, back-home, check olio. Se acceduto in modalità supervisor, il menù dovrà permettere anche l'impostazione di lampeggi frecce modalità autostrada e reset pressione gomme.

Il programma potrà essere avviato in due modalità:

- utente (lanciando solo il nome dell'eseguibile da riga di comando)
- supervisor (lanciando il nome dell'eseguibile seguito dal codice 2244).

Se avviato in modalità utente, il programma dovrà visualizzare, una riga alla volta, il seguente menù, partendo dalla prima riga e scorrendo sulle altre premendo il tasto freccia-giù/freccia-su+invio (inserire freccia-giù+invio da messaggio di riga 6 corrisponde alla visualizzazione del messaggio riga 1):

1. Setting automobile:
2. Data: 15/06/2014
3. Ora: 15:32
4. Blocco automatico porte: ON
5. Back-home: ON
6. Check olio

Se avviato in modalità supervisor, il programma dovrà visualizzare la riga 1 nel seguente modo: “Setting automobile (supervisor)”, e poter visualizzare anche le seguenti righe:

7. Frecce direzione

8. Reset pressione gomme

Ad ogni voce visualizzata, premendo i tasti freccia-destra+invio si potrà entrare nel sottomenù corrispondente, nel quale verrà visualizzato lo stato attuale del setting e data la possibilità di impostazione. Ad esempio, una volta visualizzato il menù “Blocco automatico porte: ON”, premendo il tasto freccia-destra+invio, si dovrà permettere il cambiamento del setting ON, tramite i tasti freccia-giù o freccia-su+invio. In particolare, si dovrà permettere il setting dei sottomenù 4 e 5, con possibilità di impostazione ON/OFF come nell’esempio sopra. I sottomenù 2, 3, 6, non dovranno essere implementati in questo elaborato. All’interno di un sottomenù, il solo inserimento di invio da tastiera corrisponde al ritorno al menù principale.

Se avviato in modalità supervisor, il sottomenù “Frecce direzione” dovrà visualizzare il numero dei lampeggi modalità autostrada (3 per default) con possibilità di variazione (valore minimo 2, valore massimo 5) tramite tastiera (valori inseriti fuori range corrispondono al setting del max/min valore). Nel sottomenù “Reset pressione gomme”, inserendo il carattere freccia-destra seguito da invio, il menù dovrà restituire il messaggio “Pressione gomme resettata” e tornare al menù principale.

STRUTTURA PROGETTO (parte ASM)

Ho deciso di implementare il codice assembly come una funzione main (da cui parte il programma) che si appoggia ad una funzione input che si occupa di ricevere input (appunto) da tastiera, e a funzioni per gestire i sottomenu (con distinzione tra settabili e non).

MAIN

La funzione principale del codice inizia con il controllo dei parametri a riga di comando, in particolare verifica se ne sono stati inseriti 1 o 2, nel secondo caso controlla il suo contenuto: se è 2244 allora stampa la prima riga del menu supervisor, altrimenti stampa quella del menu normale.

Il resto del codice è molto meccanico, gestisce ogni riga del menu corrispondente (supervisor o non), controllando quale freccia è stata inserita con il supporto della funzione input e agendo di conseguenza facendo una jmp alla riga precedente/successiva o chiamando una delle funzioni che gestiscono i sottomenu.

Variabili

- Una variabile stringa per ogni riga di menu supervisor e non (9 in totale)
- Una variabile long che contiene la lunghezza di ogni stringa dei menu

- Una variabile byte (trattata come booleana) che indica la presenza o meno del parametro “2244” a riga di comando, utilizzata per distinguere i casi non comuni ai due menu (in particolare riga 1 normale / 1 supervisor, 2 e 6)
- 3 variabili che contengono lo stato attuale dei menu settabili (4,5 e 7): 2 stringhe “ON”/”OFF” e 1 variabile byte inizializzata a 3
- Una variabile stringa per confrontare l’eventuale parametro passato a riga di comando con “2244”

INPUT

La funzione input ha il compito di memorizzare un input da tastiera e verificare che questo sia o meno una delle frecce interessate (SU, GIU, DX). Per farlo controlla che il primo byte sia la sequenza di escape (in ASCII 27), il secondo sia '[' (in ASCII 91) e, se sono presenti entrambi, classifica la freccia controllando che il terzo byte sia 'A', 'B' o 'C'.

La funzione non riceve nulla come parametro dal chiamante. Invece restituisce un “codice” associato a ogni input (0 = ALTRO, 1 = SU, 2 = GIU, 3 = DX).

Variabili

- Una variabile long che memorizza l’input (4 byte sono sufficienti per la memorizzazione delle frecce)

SUBMENU

La funzione submenu gestisce i sottomenu, in particolare soltanto quelli non settabili. Se da una riga di menu non settabile si entra nel sottomenu viene semplicemente ristampata la riga senza l'indice. Caso a parte la riga 8 che stampa una stringa costante (come da specifiche). Anche questa funzione si appoggia alla funzione input, qualsiasi cosa che non sia la freccia a destra causa il ritorno al menu della riga corrispondente, altrimenti viene semplicemente ristampata la stringa del sottomenu.

Al contrario della funzione input, questa non ritorna nessun valore ma riceve un parametro che indica la riga del menu che ha chiamato la funzione (0 è associato alla prima riga del menu supervisor).

Variabili

- Una variabile stringa per ogni stringa dei sottomenu non settabili (6 in totale)
- Una variabile long che contiene la lunghezza di ogni stringa

ON OFF

La funzione ON/OFF si occupa di gestire il sottomenu di 4 e 5 modificandone il valore. Confronta lo stato attuale, lo stampa, chiama la funzione input, con freccia su/giu “cambia” lo stato attuale (tra virgolette perché in realtà lo stato viene cambiato soltanto prima di uscire dalla

funzione, ho sfruttato il fatto che può essere vista come una variabile booleana, quindi una esclude l'altra), lo stampa e ricomincia il ciclo .

Come parametro dal chiamante riceve lo stato attuale, in ritorno fornisce lo stato finale che verrà settato correttamente nelle variabili dal chiamante.

Variabili

- Due variabili stringa “ON” e “OFF” per confrontare e modificare lo stato attuale

SET SETTE

La funzione set sette si occupa della gestione del sottomenu della riga 7. Stampa il valore attuale delle frecce, legge il nuovo valore da tastiera, lo setta correttamente nel valore attuale, ripete il ciclo.

Come parametro dal chiamante riceve il valore attuale delle frecce, in ritorno fornisce il valore finale

Variabili

- Una variabile che memorizza il valore attuale
- Una variabile che memorizza il nuovo valore
- Una variabile che memorizza l'andata a capo

STRUTTURA PROGETTO (parte C)

Il codice C è strutturato molto similmente a quello asm: le funzioni utilizzate sono le medesime, con lo stesso nome e lo stesso scopo per facilitare anche la comprensione del progetto nel suo complesso.

L'unica differenza sostanziale sta nella maggiore efficienza della gestione del passaggio da una riga all'altra, e ai corrispettivi sottomenu. Infatti viene tutto effettuato grazie alla variabile "code" che tiene traccia della posizione corrente (ovvero la riga del menu/sottomenu), se viene inserita la freccia su code viene diminuito di 1, se l'input è freccia giù code viene incrementato, se viene inserita la freccia a destra viene chiamata la funzione submenu passando come parametro proprio code. Chiaramente ci sono degli accorgimenti da fare nella gestione della variabile, per esempio nelle prime e ultime righe del menù non si potrà seguire la stessa logica delle altre, così come nella riga 2 e 6 (per il motivo spiegato anche per la parte di assembly).

PSEUDO-CODICE

Main

If (parametro == 2244)

 Supervisor = 1

Stampa (riga 1 supervisor)

Else

Supervisor = 0

Stampa (riga 1)

Freccia = input ()

If (freccia == SU)

Stampa (riga precedente)

Else if (freccia == GIU)

Stampa (riga successiva)

Else if (freccia == DX && riga == 1/2/3/6/8)

Submenu (riga corrente)

Else if (freccia == DX && riga == 4/5)

ON_OFF(status corrente)

Else if (freccia == DX && riga == 7)

Set_sette(valore attuale frecce)

Else

Return

SCELTE PROGETTUALI

Il codice (sia assembly che C) implementa il cruscotto della macchina in modo che esca dall'esecuzione in questi casi:

- Viene inserito un input errato (diverso dalle 3 frecce) quando ci si trova nel menu
- Viene inserito un input di dimensione troppo elevata (non più di un carattere) quando ci si trova nel sottomenu della riga 7

Invece se ci si trova in uno qualsiasi dei sottomenu, torna al menu corrispondente se:

- Viene inserito un input che non sia la freccia a destra se ci si trova nel sottomenu non settabili
- Viene inserito un input che non sia freccia su o giù se ci si trova nel sottomenu 4 o 5
- Viene inserito un input che non sia una cifra se ci si trova nel sottomenu 7

Per completezza ho deciso di “implementare” anche il sottomenu delle righe non settabili in modo che stampassero la riga stessa senza indice.

Le funzioni del codice assembly passano e ricevono i parametri inserendo i valori nei registri.