Laboratorio di Architettura degli Elaboratori

Elaborato ASM

A. A. 2014/2015

Luca Pizzini VR370826

Sommario

Descrizione del Progetto

Sviluppare un programma che controlli l'inclinazione dei flap di un Airbus A320 in base al numero e alla distribuzione dei posti a sedere dei passeggeri. L'aereo può contenere un massimo di 180 passeggeri, disposti su 6 file (A, B, C, D, E, F).
Il programma viene lanciato da riga di comando con 3 cifre come parametri:

```
./run x y z
```

x y z codificano due modalità di funzionamento:

- Controllo dinamico (3 3 2)
- Controllo emergenza (9 9 2)

Nel caso di codice errato il programma stampa un messaggio di errore e deve permettere di inserire nuovamente le 3 cifre. Nel caso di 3 tentativi errati consecutivi il programma termina con un fail.
Se il codice inserito è 3 3 2 il programma richiede di inserire il numero totale di passeggeri sull'aereo e per ogni fila. Se il controllo della coerenza del numero di passeggeri e negativo fa ripartire l'inserimento dei 7 valori, altrimenti calcola le differenze di peso tra parte sinistra e destra e imposta di conseguenza il bias per l'inclinazione dei flap.
Se il codice inserito è 9 9 2 il programma termina dopo aver stampato un messaggio di inserimento della modalità di emergenza.

Variabili Utilizzate

Ogni variabile di tipo stringa è seguita da una *variabile_len* che rappresenta la sua lunghezza.

main.s

- **count:** long utilizzato come contatore del numero di richieste consecutive di inserimento della sequenza iniziale a seguito di errore.
- **titolo:** stringa contenente il titolo del programma
- **failure:** stringa contenente un messaggio di failure da usare nel caso di 3 inserimenti di sequenze errate consecutivi
- **errato:** stringa contenente un messaggio di errore nel caso di inserimento di una sequenza sbagliata

itoa.s

- **car:** byte contenente il prossimo carattere da stampare

atoi_input.s

- **car:** byte contenente i caratteri digitati dall'utente da tastiera uno alla volta

atoi_stack.s

- **car:** byte contenente i caratteri da convertire letti dal registro eax uno alla volta

controllo_dinamico.s

- **controllo:** stringa contenente un messaggio di notifica inserimento modalità controllo dinamico
- **totale:** stringa contenente un messaggio di richiesta inserimento n° totale passeggeri a bordo
- **abc/def:** stringhe contenenti un messaggio di richiesta inserimento n° totale passeggeri per fila
- **a/b/c/d/e/f:** stringhe contenenti la lettera corrispondente alla fila richiesta per guidare l'inserimento dei valori
- **diverso:** stringa contenente un messaggio di errore in caso di n° passeggeri non coerente

- **superamento_totale:** stringa contenente un messaggio di errore nel caso il numero di passeggeri inserito sia superiore a 180
- **superamento_totale_fila:** stringa contenente un messaggio di errore nel caso il numero di passeggeri per fila inserito sia superiore a 30

controllo_emergenza

- **emergenza:** stringa contenente un messaggio di notifica inserimento modalità controllo emergenza

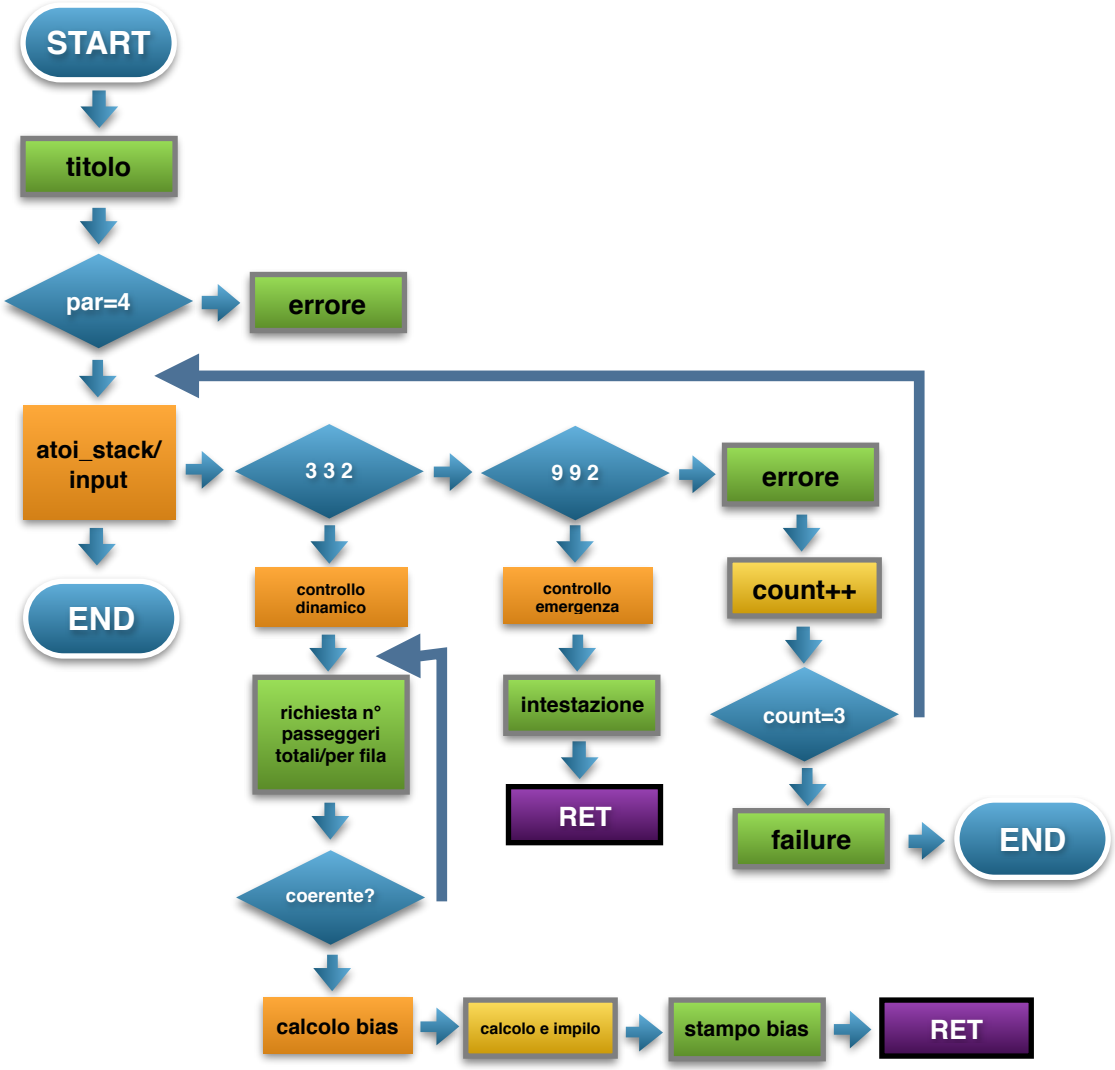
calcolo_bias

- **bias1/bias2/bias3/bias4:** stringhe che scandiscono la stampa dei valori bias calcolati

numero_negativo

- **meno:** stringa anteposta a un numero nel caso sia negativo

Diagramma di Flusso



Descrizione

- Chiamata a funzione
- System call write
- Operazioni aritmetiche
- Compare: NO → SI ↓

Modalità di Passaggio/ Restituzione dei Valori delle Funzioni Create

itoa.s
Stampa a video l'intero contenuto nel registro eax.
La funzione modifica soltanto il registro eax.

atoi_input.s
Legge input da tastiera e lo salva in eax.
La funziolne modifica soltanto il registro eax.

atoi_stack.s
Legge la stringa contenuta nell'indirizzo puntato da eax e la converte in un numero.
La funzione modifica soltanto il registro eax.

controllo_dinamico.s
Salva su stack i parametri riguardanti il numero di passeggeri a bordo letti da tastiera tramite *atoi_input*. Dopo aver controllato la coerenza dei valori chiama la funzione *calcolo_bias*.

calcolo_bias
Spila i valori salvati sullo stack da *controllo_dinamico* e calcola i valori x/2 * k necessari al calcolo del bias e li salva sullo stack.
Utilizzando i valori calcolati stampa i valori di bias richiesti.

numero_negativo
Se il valore contenuto in eax è negativo lo modifica rendendolo positivo, altrimenti non compie modifiche.

Scelte Progettuali

Controllo valori immessi
In caso di valori immessi maggiori alla capienza dell'aereo vengono stampati messaggi di errore prima di permettere il reinserimento dei dati.
In caso di non coerenza del totale dei passeggeri inserito con il valore effettivo viene stampato un messaggio di errore prima di permettere il reinserimento dei dati.

Gestione stack
I dati sui passeggeri vengono salvati su stack dalla funzione *controllo_dinamico* per poi essere facilmente utilizzabili dalla funzione *calcolo_bias*. Quest'ultima utilizza a sua volta lo stack per salvare i valori necessari al calcolo dei bias.

Divisione in più file
Il programma è stato diviso in tanti file quante sono le funzioni svolte per agevolare la fase di correzione la chiarezza dei contenuti.

Bias3 e Bias4
Essendo questi l'opposto dei valori Bias2 e Bias1 rispettivamente, non sono stati ricalcolati utilizzando ulteriore memoria. Per ottenerli è bastato negare il loro opposto.

Divisione per 2
Per effettuare la divisione per 2 richiesta nel calcolo del Bias ho effettuato uno shift a destra (sar) invece di una divisione con segno (idiv) per semplificare la gestione dei registri.