**2009학년도 1학기**

**컴퓨터 시스템 Term Project**

어셈블리언어로 만든

성적 관리 프로그램

**정보통신대학**

**컴퓨터학과**

**2000160005 한규혁**

**1. 프로그램 개발 환경**

- VisualStudio 2005 MASM

- Petium4 2.4 / Windows XP

**2. 프로그램 실행 방법**

1) Project.exe 파일을 실행한다

2) 이름을 입력한다. (8자 이상은 짤림)

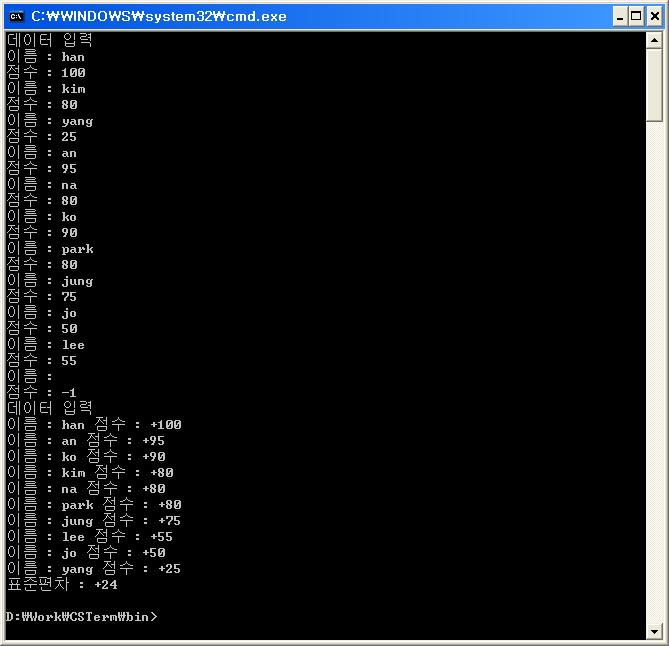
3) 점수를 입력한다. (100점 초과는 다시 입력)

4) 학생입력이 끝났으면 이름 공백, 점수 -1을 입력한다.

(최대 100명까지 입력가능)

5) 정렬된 값과 표준편차 값을 확인한다.

**3. 프로그램 결과**



<프로그램 실행 화면 - 10개의 셈플 데이터를 입력한 결과 화면>

**4. 프로그램 순서도**

<프로그램 순서도>

**5. 결론**

본 프로그램 작성시 가장 주의하였던 부분이 상위레벨 언어인 C언어나 Java언어 스타일과 비슷한 IF, WHILE등의 Directive의 사용을 지양하는 것이었다. 이는 본인이 상위 언어로 코딩할 때, 간혹 디버깅 정보가 남지 않아 어셈블리어로 디버깅해야 하는 경우가 있었는데, 이때 보여지는 코드는 Directive가 제외된 상태의 코드이므로, 순수 어셈블리 언어에 익숙해지기 위함이었다. 본 프로그램 작성으로 레지스터 사용법과 기본적인 키워드인 loop, jmp, cmp, call, mov, add, sub, mul, div만을 가지고 코딩할수 있었고, 이러한 기본적인 문법에 대해서는 확실하게 익힐 수 있는 계기가 된 거 같다.

**Appendix. 프로그램 소스 코드**

1) 메인 프로시저

|  |
| --- |
| main PROC  call Clrscr    ;데이터입력  call InputData    ;데이터입력확인  cmp dataSize, 0  jne PROCESS  exit  PROCESS:  ;데이터크기확인  cmp dataSize, 1  jbe PRINT ; datasize <= 1 ?    ;소팅  call BubbleSort  ;분산계산  call Deviation  PRINT:  ;데이터출력  call PrintData    exit  main ENDP |

2) 입력 프로시저

|  |
| --- |
| InputData PROC USES eax ecx edi edx esi  ;입력메세지출력  mov edx, offset inputText  call writestring  call crlf    ;루프의카운터  mov ecx, 100  dec ecx    ;초기값  mov edi, offset studentdata  mov dataSize, 0    INPUT:  push ecx    ;이름입력메세지를띄우고값을읽어옴  mov edx, offset nameText  call writestring  mov edx, offset tempinputName  mov ecx, (Sizeof tempinputName)  call readstring  INPUTPOINT:  ;점수입력메세지를띄우고값을읽어옴  mov edx, offset pointText  call writestring  call readint  mov tempinputpoint, eax    ;점수가-1 이면입력끝  cmp eax, -1  jne SAVE  pop ecx  ret  SAVE:  ;점수가100보다크면다시입력  cmp tempinputpoint, 100 ;tempinputpoint > 100 ?  ja INPUTPOINT  inc dataSize    ;이름저장  mov esi, offset tempinputName  invoke Str\_copy, esi, edi    ;점수저장  mov eax, tempinputpoint  mov (StudentInfo ptr [edi]).point, al    ;넥스트  add edi, structsize  ;아니면계속루프  pop ecx  loop INPUT  ret  InputData ENDP |

3) 정렬 프로시저

|  |
| --- |
| BubbleSort PROC USES eax ecx edi  SortIntPart:  ;루프의카운터  mov ecx, dataSize  ;dec ecx  ;버플소팅시작  StartSort:  push ecx  mov edi, offset studentData  Compare:  ;원소2개잡음  mov bl, [edi+pointpos]  mov dl, byte ptr [edi+structsize+pointpos]    ;비교  cmp bl, dl  jge noChange  ;점수교환  xchg bl, dl  mov [edi+pointpos], bl  mov byte ptr [edi+structsize+pointpos],dl    ;이름교환  push eax  INVOKE Str\_copy, ADDR (Studentinfo ptr [edi]).stuName, ADDR tempinputname  INVOKE Str\_copy, ADDR (Studentinfo ptr [edi+structsize]).stuName, ADDR (Studentinfo ptr [edi]).stuName  INVOKE Str\_copy, ADDR tempinputname, ADDR (Studentinfo ptr [edi+structsize]).stuName  pop eax    noChange:  ;넥스트  add edi, structsize  loop Compare    pop ecx  Loop StartSort    ret  BubbleSort ENDP |

4) 표준편차 계산 프로시저

|  |
| --- |
| Deviation PROC USES eax ecx edi edx  ;지역변수  LOCAL SQUSUM:DWORD  LOCAL SUM:DWORD  LOCAL RES:DWORD  LOCAL SQRT:DWORD  mov SQRT, 0  ;점수제곱의합구하기  ;초기값  mov ecx, dataSize  mov edi, offset studentData  mov edx, 0  mov eax, 0  CALCULATION1:  push ecx    ;Σi=1,N xi2 계산  mov al, (StudentInfo ptr [edi]).point  mov bl, al  mul bl  add edx, eax    pop ecx    ;넥스트  add edi, structsize  loop CALCULATION1    mov SQUSUM, edx    ;점수의합구하기  ;초기값  mov ecx, dataSize  mov edi, offset studentData  mov edx, 0  mov eax, 0  CALCULATION2:  push ecx    ;Σi=1,N xi 계산  mov al, (StudentInfo ptr [edi]).point  add edx, eax    pop ecx    ;넥스트  add edi, structsize  loop CALCULATION2    mov SUM, edx    CALCULATION3:  ;σ= [{Σi=1,N xi2 - (Σi=1,N xi)2 /N} /(N-1) ] 계산  ;점수의합의제곱  mov eax, SUM  mov ebx, eax  mul ebx    ;점수의합의제곱에평균  mov ebx, dataSize  div ebx    ;중괄호내부계산  sub SQUSUM, eax    ;대괄호내부계산  mov eax, SQUSUM  mov ebx, dataSize  sub ebx, 1  div ebx    mov RES, eax    ;제곱근계산  ;초기값  mov edi, 1 ; n = 0  mov edx, 0 ; sigma  mov eax, 0  CALCULATION4:  ;Σ(2n-1)  mov eax, 2  mov ebx, edi  mul ebx  sub eax, 1  add SQRT, eax    ;  mov edx, SQRT  cmp edx, RES ;edx >= RES ?  jae CALCULATION5    ;  inc edi  mov ecx, 2  loop CALCULATION4  CALCULATION5:  ;표준편차값  mov deviationValue, edi    ret    Deviation ENDP |

5) 출력 프로시저

|  |
| --- |
| PrintData PROC USES eax ecx edi edx  ;출력메세지출력  mov edx, offset inputText  call writestring  call crlf    ;루프의카운터  mov ecx, dataSize  ;dec ecx    ;초기값  mov edi, offset studentData    ;출력시작  Print:  push ecx    ;이름메세지와출력  mov edx, offset nameText  call writestring  mov edx, edi  call writestring  mov edx, offset spaceText  call writestring    ;점수메시지와출력  mov edx, offset pointText  call writestring  mov al, [edi+pointpos]  call writeint  call crlf    ;넥스트  add edi, structsize    pop ecx  Loop Print    ;표준편차메시지와출력  mov edx, offset deviationText  call writestring  mov eax, deviationValue  call writeint  call crlf    ret    PrintData ENDP |