**תרגיל בית 4:**

מגישים:

איילון בן סימון – 312162951

סער ויקטור – 312392822

**קוד תרגיל 1:**

;targil4.asm

;find\_delta(double (\*fp)(double), double x0,double epsilon, double range);

; BP+4 BP+6 BP+14 BP+22

.MODEL **SMALL**

.STACK 100h

.DATA

delta DQ **?**

delta1 DQ 0

delta2 DQ 0

two DQ 2.0

num DQ **?**

temp DQ **?**

.CODE

.386

.387

\_find\_delta PROC **NEAR**

PUBLIC \_find\_delta

;save registers

**PUSH** **BP**

**MOV** **BP,SP**

**FLDZ** ;st(0)=0

**FSTP** delta1 ;delta=0

**FLD** **QWORD** PTR **[BP+**6**]** ;st(0)=x0

**FDIV** two ;st(0)=st(0)/2

**FABS** ;st(0)=ABS(st(0))

**FST** delta2 ;delta2=st(0)

**FLD** delta1 ;st(1)=delta2,st(0)=delta1

L1**:**

**FSUB** ;st(0)=delta2-delta1

**FLD** **QWORD** PTR **[BP+**22**]**;st(0)=range,st(1)=delta2-delta1

**FCOMPP** ;range-(delta2-delta1)

**FSTSW** **AX**

**SAHF**

**JA** end1

**FLD** delta2 ;st(0)=delta2

**FLD** delta1 ;st(1)=delta2,st(0)=delta1

**FADD** ;st(0)=delta1+delta2

**FDIV** two ;st(0)=(delta1+delta2)/2

**FST** delta ;delta=(delta1+delta2)/2,st(0)=delta

**FLD** **QWORD** PTR **[BP+**6**]** ;st(0)=x0,st(1)=delta

**FADD** ;st(0)=x0+delta

**FST** temp

**PUSH** temp

**CALL** **[BP+**4**]** ;call function fp (sin(x0+delta))

**POP** temp

**FSTP** **st(**1**)** ;st(0)=sin(x0+delta),st(1)=empty

**FLD** **QWORD** PTR **[BP+**6**]** ;st(0)=X0,st(1)=sin(x0+delta)

**FST** temp

**PUSH** temp

**CALL** **[BP+**4**]** ;call function fp (sin(x0))

**FSTP** **st(**1**)** ;st(0)=sin(x0),st(1)=sin(x0+delta),st(2)=empty

**POP** temp

**FSUB** ;st(0)=sin(x0+delta)-sin(x0)

**FABS** ;st(0)=ABS(st(0))

**FLD** **QWORD** PTR **[BP+**14**]**;st(0)=epsilon, st(1)=ABS(sin(x0+delta)-sin(x0))

**FCOMPP** ;epsilon-ABS(sin(x0+delta)-sin(x0))

**FSTSW** **AX**

**SAHF**

**JB** case

**FLD** delta ;st(0)=delta

**FST** delta1 ;delta1=delta "or" st(0)

**FSTP** temp

**FLD** delta2 ;st(0)=delta2

**FLD** delta1 ;st(0)=delta1 , st(1)=delta2

**JMP** L1

case**:**

**FLD** delta ;st(0)=delta

**FST** delta2 ;delta2=delta/st(0)

**FSTP** temp

**FLD** delta2 ;st(0)=delta2

**FLD** delta1 ;st(0)=delta1 , st(1)=delta2

**JMP** L1

end1**:**

**FLD** delta1 ;st(0)=delta1

**FLD** delta2 ;st(0)=delta2,st(1)=delta1

**FADD** ;st(0)=delta1+delta2, st(1)=empty

**FDIV** two ;st(0)=(delta1+delta2)/2

**POP** **BP**

**RET**

\_find\_delta ENDP

END

**פלט תרגיל 1:**

