

Procedures for Hungarian Multiplication

Type HUNGARY to begin

```

to hungary
clearpage
initialize
enter
test
math.figure
final
end

to clearpage
rg ht
ct cc
end

to initialize
make "factor1 0
make "factor2 0
make "count1 0
make "count2 0
end

to enter
print [Enter your first
factor.]
make "factor1 readnumber
print [Enter your second
number.]
make "factor2 readnumber
print []
end

to readnumber
output first readlist
end

to math.figure
if :factor1 < 1 [stop]
print (sentence [First factor
is] :factor1 [; the second]
:factor2)
make "factor1 intquotient
:factor1
make "factor2 :factor2 * 2
test
math.figure
end

to test
ifelse even? :factor1
[make "count1 :count1 +
:factor2 insert [-]]

```

```

[make "count2 :count2 +
:factor2 insert [*]],
end

to even? :number
output member? (last :number)
[0 2 4 6 8]
end

to final
print (sentence [The product
of your problem is] :count2)
print []
print [The Hungarians added
all second factors in the
lines above that begin with
asterisks.]
continue
end

to intquotient :number
output int :number / 2
end

to continue
cc show [To continue, type the
command HUNGARY and press
RETURN...]
print []
end

```

Procedures for Egyptian Multiplication

Type CHOOSE to begin

```

to choose
clearpage
repeat 4 [print []]
insert (sentence [To half your
first factor and double the
second, reply with (Y).] char
13)
insert [To double your first
factor and half the second,
reply with (N).]
ifelse readchar = "Y [Egypt]
[Egypt1]
end

to Egypt
clearpage initialize
enter test
math.figure
final.e
end

```

```

to Egypt1
clearpage initialize
enter test1
math.figure1
final1
end

to math.figure1
if :factor2 < 1 [stop]
print (sentence [First factor
is] :factor1 [; the second]
:factor2)
make "factor1 intquotient
:factor1
make "factor2 :factor2 * 2
test1
math.figure1
end

to test1
ifelse even? :factor2
[make "count1 :count1 +
:factor1 insert [-]]
[make "count2 :count2 +
:factor1 insert [*]]
end

to final.e
print []
print sentence [The product of
your problem is] :count2
print []
print [The Egyptians added all
second factors in lines that
begin with asterisks.]
continue.e
end

to final1
print []
print sentence [The product of
your problem is] :count2
print []
print [The Egyptians added all
first factors in lines that
begin with asterisks.]
continue.e
end

to continue.e
cc show [To continue, type the
command CHOOSE and press
RETURN...]
print []
end

```

