Topic VOD

Recursive Functions

Readings: (Section 2.8)

Non-Leaf Procedures

Procedures that call other procedures

- For nested calls, callee needs to:
 - Save on the stack:
 - ra
 - callee-saved registers
 - Restore values from the stack before returning

Calling Itself

```
int fact(int n){
    if (n < 1)
        return 1;
    else
        return (n * fact(n - 1));
}</pre>
```

Step-by-step creation of the assembly code for fact

```
int fact(int n){
    if (n < 1)
        return 1;
    else
        return (n * fact(n - 1));
}</pre>
```

```
save ra;
if (n < 1){
        restore ra;
        return 1;
}
else{
        tA ← n * fact(n - 1);
        restore ra;
        return tA;
}</pre>
```

```
save ra;
if (n < 1){
    restore ra;
    a0 ← 1;
    return a0;
}
else{
    tA ← n * fact(n - 1);
    restore ra;
    return tA;
}</pre>
```

```
save ra;
if (n < 1){
       restore ra;
        a0 ← 1;
        return a0;
else{
        a0 \leftarrow n - 1;
       tA \leftarrow n * fact(a0);
        restore ra;
        return tA;
```

```
n is also in a0
```

```
save ra;
if (n < 1){
        restore ra;
        a0 ← 1;
        return a0;
else{
       a0 \leftarrow n - 1;
       tA \leftarrow n * fact(a0);
        restore ra;
        return tA;
```

```
save ra;
if (n'< 1){
       restore ra;
       a0 ← 1;
       return a0;
else{
       a0 ← n - 1;
       a0 ← fact(a0);
       tA \leftarrow n * a0;
       restore ra;
       return\tA;
```

```
save ra, a0;
if (n < 1){
      restore ra, a0;
      a0 ← 1;
      return a0;
else{
      a0 ← n - 1;
      a0 ← fact(a0);
      restore ra, a0;
      tA ← n * a0;
      return tA;
```

n is used here

```
save ra, a0;
if (n < 1){
      restore ra, a0;
      a0 ← 1;
      return a0;
else{
      a0 ← n - 1;
      a0 ← fact(a0);
      restore ra, a0;
      tA ← n * a0;
      return tA;
```

```
save ra, a0;
if (a0 < 1){
        restore ra, a0;
        a0 ← 1;
        return a0;
     n-1
else{
       a0 \leftarrow a0 - 1;
        a0 \leftarrow fact(a0);
        restore ra, a0;
        tA \leftarrow a0 * a0;
        return tA;
```

What is wrong with this?

This multiplication is n*n
It should be n*fact(n-1)

```
save ra, a0;
if (a0 < 1){
       restore ra, a0;
       a0 ← 1;
       return a0;
else{
       a0 \leftarrow a0 - 1;
       a0 ← fact(a0);
       t0 ← a0;
       restore ra, a0;
       a0 ← a0 * t0;
       return a0;
```

This is n This is fact(n-1) This a0 is n

This a0 is fact(n-1)

```
save ra, a0;
if (a0 < 1){
       restore ra, a0;
       a0 \leftarrow 1;
       return a0;
else{
       a0 \leftarrow a0 - 1;
       a0 ← fact(a0);
       t0 ← a0;
       restore ra, a0;
       a0 ← a0 * t0;
       return a0;
```

```
sp \leftarrow sp - 8;
M[sp + 4] \leftarrow ra;
M[sp] \leftarrow a0;
if (a0 < 1){
         restore ra, a0;
          a0 \leftarrow 1;
          return a0;
else{
          a0 \leftarrow a0 - 1;
          a0 \leftarrow fact(a0);
          t0 ← a0;
          a0 \leftarrow M[sp];
          ra \leftarrow M[sp + 4];
          sp \leftarrow sp + 8;
          a0 \leftarrow a0 * t0;
          return a0;
```

```
sp \leftarrow sp - 8;
M[sp + 4] \leftarrow ra;
M[sp] \leftarrow a0;
if (a0 < 1){
          <u>ra ← M[sp + 4]</u>
          a0 ← M[sp];
          sp \leftarrow sp + 8;
          a0 \leftarrow 1;
          return a0;
else{
          a0 \leftarrow a0 - 1;
          a0 ← fact(a0);
          t0 ← a0;
          a0 \leftarrow M[sp];
          ra \leftarrow M[sp + 4];
          sp \leftarrow sp + 8;
          a0 \leftarrow a0 * t0;
          return a0;
```

Recursive Procedure

```
sp \leftarrow sp - 8;
M[sp + 4] \leftarrow ra;
M[sp] \leftarrow a0;
if (a0 < 1){
         sp \leftarrow sp + 8;
         a0 ← 1;
         return a0;
else{
         a0 \leftarrow a0 - 1;
         a0 ← fact(a0);
         t0 ← a0;
         a0 \leftarrow M[sp];
         ra \leftarrow M[sp + 4];
         sp \leftarrow sp + 8;
         a0 \leftarrow a0 * t0;
         return a0;
```

```
int fact(int n){
    if (n < 1)
        return 1;
    else
        return (n * fact(n - 1));
}</pre>
```

Recursive Procedure

```
sp \leftarrow sp - 8;
M[sp + 4] \leftarrow ra;
M[sp] \leftarrow a0;
if (a0 < 1){
         sp \leftarrow sp + 8;
         a0 \leftarrow 1;
          return a0;
else{
         a0 \leftarrow a0 - 1;
         a0 ← fact(a0);
         t0 ← a0;
          a0 \leftarrow M[sp];
          ra \leftarrow M[sp + 4];
          sp \leftarrow sp + 8;
          a0 \leftarrow a0 * t0;
          return a0;
```

```
if (n < 1)
                                       return 1;
                                else
                                       return (n * fact(n - 1));
fact:
   addi
                            # make room in stack for 2 more items
              sp, sp, -8
              ra, 4(sp)
                            # save the return address
   SW
              a0, 0(sp)
                            # save the argument n
   SW
             t1, 1 # t1 <- 1
   li
                            # if n \ge 1, go to L1
   bge
              a0, t1, L1
   addi
              a0, zero, 1
                          # return 1
   addi
              sp, sp, 8
                            # pop two items from the stack
   jalr
              zero, ra, 0
                            # return to the instruction after jal
L1: addi
              a0, a0, -1
                            # subtract 1 from the argument
              ra, fact
   jal
                            # jump to fact and save position to ra
             t0, a0
                            # copy the return value; don't overwrite
   ΜV
              a0, 0(sp)
                            # just returned from jal, restore n
   lw
              ra, 4(sp)
                            # restore the return address
   lw
   addi
              sp, sp, 8
                            # pop two items from the stack
   mul
              a0, a0, t0
                            # return n * fact(n - 1)
   jalr
              zero, ra, 0
                           # return to the caller
```

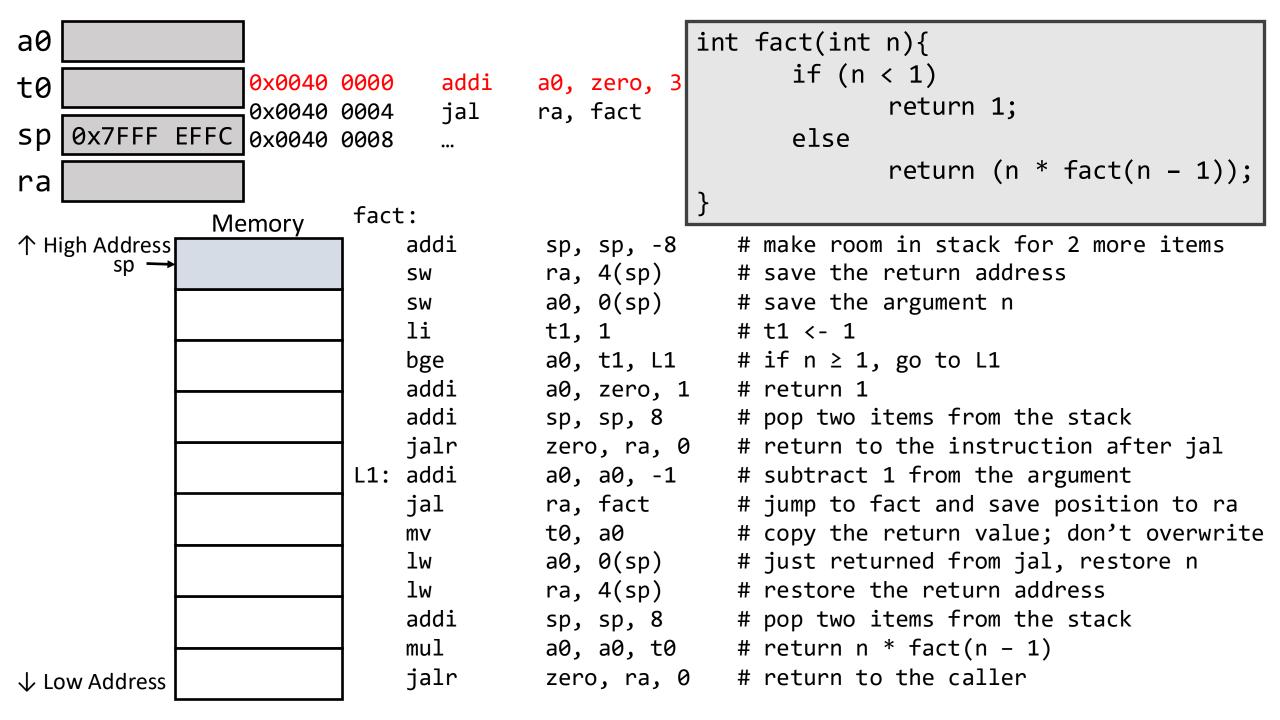
int fact(int n){

```
fact:
int fact(int n){
                                                       addi
                                                                   sp, sp, -8
                                                                   ra, 4(sp)
  save ra;
                                                       \mathsf{SW}
                                                                   a0, 0(sp)
  id save ra:
                                                       SW
                                                       li
                                                                   t1, 1
     if (sp \leftarrow sp - 8)
                                                       bge
                                                                   a0, t1, L1
                                        - 1));
           M[sp + 4] \leftarrow ra;
                                                       addi
                                                                   a0, zero, 1
          M[sp] \leftarrow a0;
                                                       addi
                                                                   sp, sp, 8
           if (a0 < 1){
                                                       jalr
                                                                   zero, ra, 0
                  sp \leftarrow sp + 8;
                                                  L1: addi
                                                                   a0, a0, -1
                 a0 ← 1;
     else
                                                                   ra, fact
                                                       jal
                   return a0;
                                                                   t0, a0
                                                       ΜV
                                                       lw
                                                                   a0, 0(sp)
           else{
                                                       lw
                                                                   ra, 4(sp)
                                                       addi
                   a0 \leftarrow a0 - 1;
                                                                   sp, sp, 8
                                                       mul
                                                                   a0, a0, t0
                   a0 ← fact(a0);
                                                       jalr
                                                                   zero, ra, 0
                   t0 ← a0;
                   a0 \leftarrow M[sp];
                   ra \leftarrow M[sp + 4];
                                           Recap
                   sp \leftarrow sp + 8;
                   a0 \leftarrow a0 * t0;
                   return a0;
```

The simulation will not track t1 because it is only stores the value 1 for these instructions:

 $n \ge 1 \equiv n > 0$ if n is an integer. The two instructions above could be replaced with:

blt zero, a0, L1 # if
$$n > 0$$
, go to L1

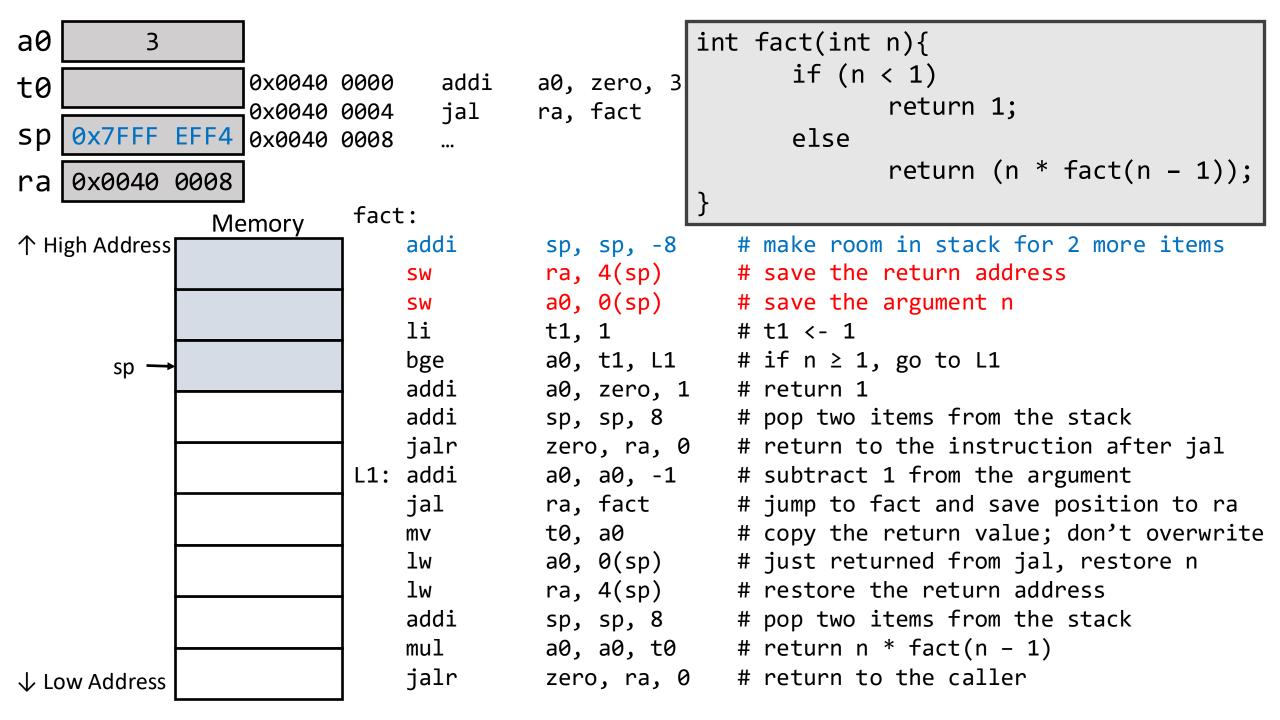


```
a0
                                                  int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                               addi
                                     a0, zero, 3
t0
                                                                 return 1;
                 0x0040 0004
                                      ra, fact
                               jal
   0x7FFF EFFC 0x0040 0008
sp
                                                         else
                                                                 return (n * fact(n - 1));
ra
                         fact:
              Memory
个 High Address [
                             addi
                                                     # make room in stack for 2 more items
                                       sp, sp, -8
       sp
                                       ra, 4(sp)
                                                     # save the return address
                             SW
                                       a0, 0(sp)
                                                     # save the argument n
                             SW
                             li
                                       t1, 1
                                              # t1 <- 1
                                                     # if n \ge 1, go to L1
                             bge
                                       a0, t1, L1
                             addi
                                       a0, zero, 1
                                                    # return 1
                             addi
                                       sp, sp, 8
                                                     # pop two items from the stack
                             jalr
                                       zero, ra, 0
                                                     # return to the instruction after jal
                         L1: addi
                                       a0, a0, -1
                                                     # subtract 1 from the argument
                                       ra, fact
                             jal
                                                     # jump to fact and save position to ra
                                       t0, a0
                                                     # copy the return value; don't overwrite
                             ΜV
                                       a0, 0(sp)
                             lw
                                                     # just returned from jal, restore n
                                       ra, 4(sp)
                                                     # restore the return address
                             lw
                             addi
                                       sp, sp, 8
                                                     # pop two items from the stack
                             mul
                                       a0, a0, t0
                                                     # return n * fact(n - 1)
                             jalr
                                       zero, ra, 0
                                                     # return to the caller

↓ Low Address
```

```
a0
                                                  int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                               addi
                                      a0, zero, 3
t0
                                                                 return 1;
                 0x0040 0004
                               jal
                                       ra, fact
   0x7FFF EFFC 0x0040 0008
sp
                                                          else
                                                                 return (n * fact(n - 1));
ra | 0x0040 0008
                         fact:
              Memory
个 High Address [
                             addi
                                                     # make room in stack for 2 more items
                                       sp, sp, -8
       sp
                                       ra, 4(sp)
                                                     # save the return address
                             SW
                                       a0, 0(sp)
                                                     # save the argument n
                             SW
                             li
                                       t1, 1
                                              # t1 <- 1
                                                     # if n \ge 1, go to L1
                             bge
                                       a0, t1, L1
                             addi
                                       a0, zero, 1
                                                     # return 1
                             addi
                                       sp, sp, 8
                                                     # pop two items from the stack
                             jalr
                                       zero, ra, 0
                                                     # return to the instruction after jal
                         L1: addi
                                       a0, a0, -1
                                                     # subtract 1 from the argument
                                       ra, fact
                             jal
                                                      # jump to fact and save position to ra
                                       t0, a0
                                                     # copy the return value; don't overwrite
                             ΜV
                                       a0, 0(sp)
                             lw
                                                     # just returned from jal, restore n
                                       ra, 4(sp)
                             lw
                                                     # restore the return address
                             addi
                                       sp, sp, 8
                                                     # pop two items from the stack
                             mul
                                       a0, a0, t0
                                                     # return n * fact(n - 1)
                             jalr
                                       zero, ra, 0
                                                     # return to the caller

↓ Low Address
```



```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                                addi
                                      a0, zero, 3
t0
                                                                 return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                          else
                                                                 return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                       sp, sp, -8
                                                      # save the return address
                                       ra, 4(sp)
                             SW
                                       a0, 0(sp)
                                                      # save the argument n
                             SW
            0x0040 0008
                             li
                                       t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                                       a0, t1, L1
                             bge
       sp
                             addi
                                                      # return 1
                                       a0, zero, 1
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                       zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                       a0, a0, -1
                                                      # subtract 1 from the argument
                                       ra, fact
                             jal
                                                      # jump to fact and save position to ra
                                       t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                       a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                       ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                       a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                       zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                                addi
                                      a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                          else
                                                                  return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                       t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                 3
       sp
                             addi
                                                      # return 1
                                        a0, zero, 1
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                       ra, fact
                             jal
                                                      # jump to fact and save position to ra
                                       t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

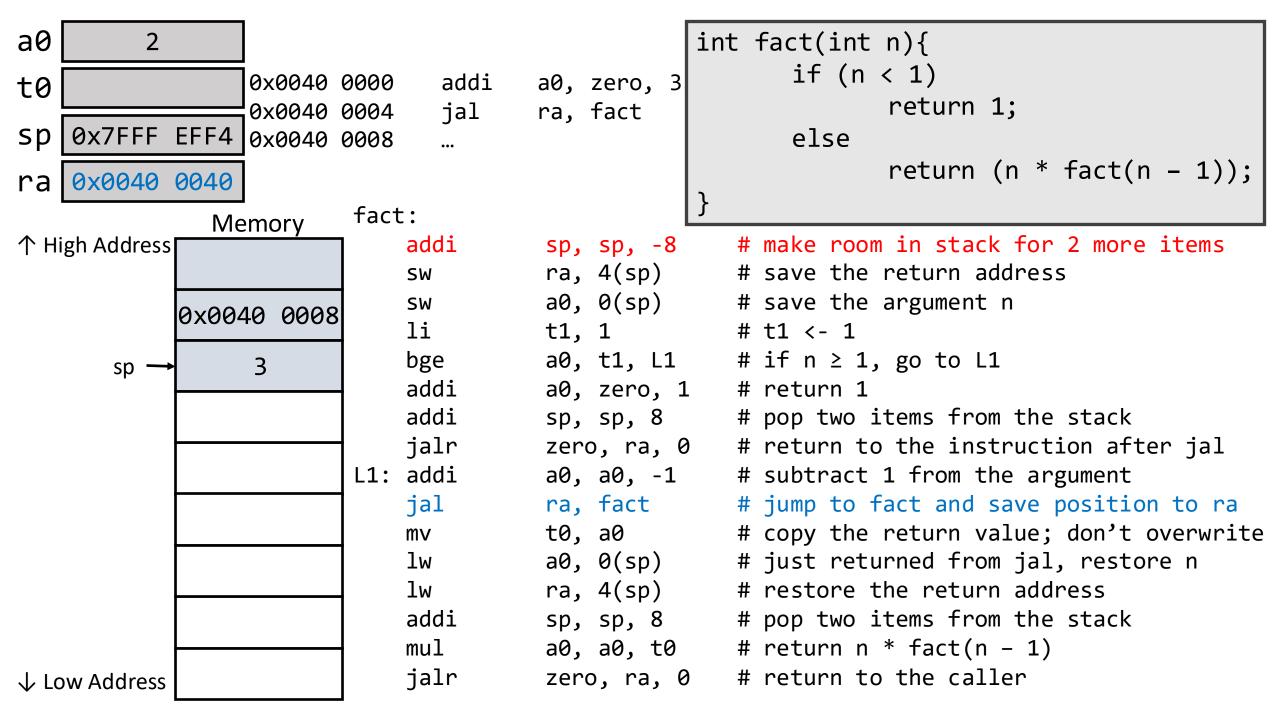
↓ Low Address
```

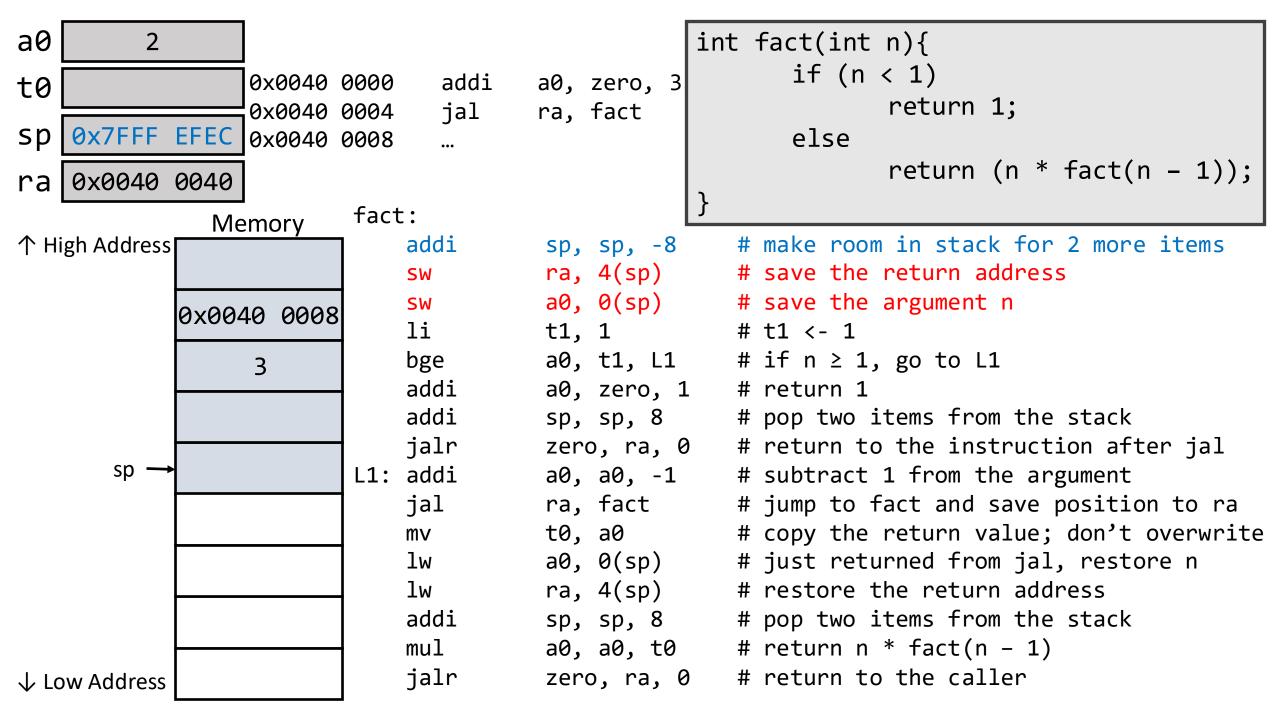
```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                               addi
                                      a0, zero, 3
t0
                                                                 return 1;
                 0x0040 0004
                                       ra, fact
                               jal
   0x7FFF EFF4 0x0040 0008
sp
                                                          else
                                                                 return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                       sp, sp, -8
                                       ra, 4(sp)
                                                      # save the return address
                             SW
                                       a0, 0(sp)
                                                     # save the argument n
                             SW
            l0x0040 0008l
                             li
                                       t1, 1
                                               # t1 <- 1
                                       a0, t1, L1
                                                     # if n \ge 1, go to L1
                             bge
                 3
       sp
                             addi
                                                     # return 1
                                       a0, zero, 1
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                       zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                       a0, a0, -1
                                                      # subtract 1 from the argument
                                       ra, fact
                             jal
                                                      # jump to fact and save position to ra
                                       t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                       a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                       ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                       a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                       zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                  int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                               addi
                                      a0, zero, 3
t0
                                                                 return 1;
                 0x0040 0004
                                       ra, fact
                               jal
   0x7FFF EFF4 0x0040 0008
sp
                                                          else
                                                                 return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                       sp, sp, -8
                                       ra, 4(sp)
                                                      # save the return address
                             SW
                                       a0, 0(sp)
                                                     # save the argument n
                             SW
            l0x0040 0008l
                             li
                                       t1, 1
                                               # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                       a0, t1, L1
                 3
       sp
                             addi
                                                     # return 1
                                       a0, zero, 1
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                       zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                       a0, a0, -1
                                                      # subtract 1 from the argument
                             jal
                                       ra, fact
                                                      # jump to fact and save position to ra
                                       t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                       a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                       ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                       sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                       a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                       zero, ra, 0
                                                      # return to the caller

↓ Low Address
```





```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                                addi
                                      a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                          else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
                             addi
                                                      # return 1
                                        a0, zero, 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                      # jump to fact and save position to ra
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                                addi
                                      a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                          else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                        a0, t1, L1
                                                      # if n \ge 1, go to L1
                             bge
                  3
                             addi
                                                      # return 1
                                        a0, zero, 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                                      # return to the instruction after jal
                                        zero, ra, 0
       sp
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                             jal
                                        ra, fact
                                                      # jump to fact and save position to ra
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                          if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                          else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                   # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                             jal
                                        ra, fact
                                                      # jump to fact and save position to ra
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                          else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                      # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
         0
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                      a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                        a0, t1, L1
                                                      # if n \ge 1, go to L1
                             bge
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                                      # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                             jal
                                        ra, fact
                                                      # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                             jal
                                        ra, fact
                                                      # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                             lw
                                                      # restore the return address
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             jalr
                                        zero, ra, 0
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
         0
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFDC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                        a0, t1, L1
                                                      # if n \ge 1, go to L1
                             bge
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                      # just returned from jal, restore n
                                        ra, 4(sp)
                                                      # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
            0x0040 0040
                             mul
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                 0
                             jalr
                                        zero, ra, 0
                                                      # return to the caller
       sp →

↓ Low Address
```

```
a0
                                                   int fact(int n){
         0
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFDC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                        a0, t1, L1
                                                       # if n \ge 1, go to L1
                             bge
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
            10x0040 00401
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                       # just returned from jal, restore n
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            10x0040 00401
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                 0
                             jalr
                                        zero, ra, 0
                                                       # return to the caller
       sp —

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFDC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
            10x0040 00401
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                                        a0, a0, -1
                         L1: addi
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                             lw
                                                       # just returned from jal, restore n
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            10x0040 00401
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                 0
                             jalr
                                        zero, ra, 0
                                                      # return to the caller
       sp —

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                      # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                                                      # restore the return address
                             lw
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                                      # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                                                      # restore the return address
                             lw
                                        sp, sp, 8
                             addi
                                                      # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                      # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                                                      # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                      # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                      # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                      # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                      # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                      # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                      # subtract 1 from the argument
                                        ra, fact
                             jal
                                                      # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                      # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                      # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                                                      # restore the return address
                             lw
                             addi
                                                      # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                      # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFE4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra | 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            l0x0040 0040l
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
       sp
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                                       # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                             mul
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                                       # return to the instruction after jal
                                        zero, ra, 0
       sp
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                              lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                                                       # return n * fact(n - 1)
                                        a0, a0, t0
                             mul
                             jalr
                                        zero, ra, 0
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                                        sp, sp, 8
                             addi
                                                       # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                             jalr
                                        zero, ra, 0
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                      # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                      # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                                       # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                      # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFEC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra | 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                  3
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            10x0040 00401
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
       sp
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                                       # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
       sp
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                             mul
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                   return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                           else
                                                                   return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
               Memory
个 High Address 「
                              addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                              SW
                                        a0, 0(sp)
                                                       # save the argument n
                              SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
       sp
                              addi
                                        a0, zero, 1
                                                       # return 1
                              addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                              jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                              jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                              lw
                                        ra, 4(sp)
                                                       # restore the return address
                              lw
                              addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                                                       # return n * fact(n - 1)
                                        a0, a0, t0
                             mul
                              jalr
                                        zero, ra, 0
                                                       # return to the caller

    ↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
       sp
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                                       # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                                        sp, sp, 8
                             addi
                                                       # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                             jalr
                                        zero, ra, 0
                                                       # return to the caller

↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0040
                         fact:
              Memory
个 High Address 「
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
       sp
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                                       # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

    ↓ Low Address
```

```
a0
                                                    int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                   return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFF4 0x0040 0008
sp
                                                           else
                                                                   return (n * fact(n - 1));
ra | 0x0040 0008
                         fact:
               Memory
个 High Address 「
                              addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
                                        ra, 4(sp)
                                                       # save the return address
                              SW
                                        a0, 0(sp)
                                                       # save the argument n
                              SW
            l0x0040 0008l
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                  3
       sp
                              addi
                                        a0, zero, 1
                                                       # return 1
                              addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                              jalr
                                        zero, ra, 0
                                                       # return to the instruction after jal
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                              jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                              lw
                                        ra, 4(sp)
                                                       # restore the return address
                              lw
                              addi
                                                       # pop two items from the stack
                                        sp, sp, 8
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                                        zero, ra, 0
                              jalr
                                                       # return to the caller

    ↓ Low Address
```

```
a0
                                                   int fact(int n){
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFFC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address [
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
       sp
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            0x0040 0008
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                                       # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                             lw
                                                       # restore the return address
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                             mul
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                                        zero, ra, 0
                             jalr
                                                       # return to the caller

    ↓ Low Address
```

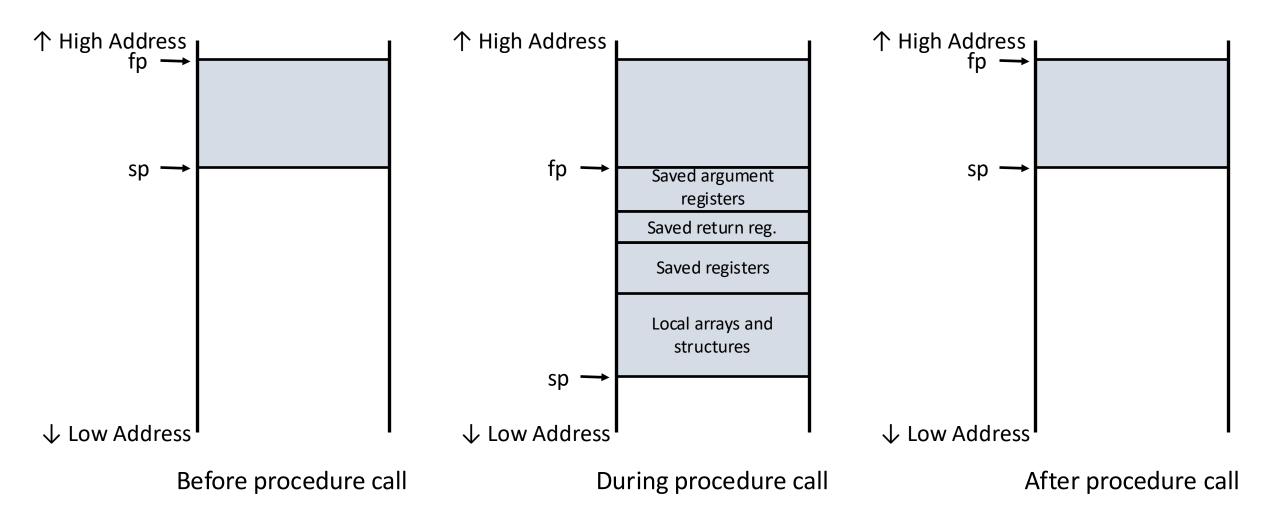
```
a0
                                                    int fact(int n){
         6
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                        ra, fact
                                jal
   0x7FFF EFFC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address [
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
       sp
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            0x0040 0008
                             li
                                        t1, 1
                                                       # t1 <- 1
                                                       # if n \ge 1, go to L1
                             bge
                                        a0, t1, L1
                             addi
                                        a0, zero, 1
                                                       # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                                       # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                              lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                             addi
                                        sp, sp, 8
                                                       # pop two items from the stack
            0x0040 0040
                                                       # return n * fact(n - 1)
                                        a0, a0, t0
                             mul
                             jalr
                                        zero, ra, 0
                                                       # return to the caller

    ↓ Low Address
```

```
a0
                                                   int fact(int n){
         6
                                                           if (n < 1)
                 0x0040 0000
                                addi
                                       a0, zero, 3
t0
                                                                  return 1;
                 0x0040 0004
                                       ra, fact
                                jal
   0x7FFF EFFC 0x0040 0008
sp
                                                           else
                                                                  return (n * fact(n - 1));
ra 0x0040 0008
                         fact:
              Memory
个 High Address [
                             addi
                                                       # make room in stack for 2 more items
                                        sp, sp, -8
       sp
                                        ra, 4(sp)
                                                       # save the return address
                             SW
                                        a0, 0(sp)
                                                       # save the argument n
                             SW
            0x0040 0008
                             li
                                        t1, 1
                                                      # t1 <- 1
                                                       # if n \ge 1, go to L1
                                        a0, t1, L1
                             bge
                             addi
                                        a0, zero, 1
                                                      # return 1
                             addi
            0x0040 0040
                                        sp, sp, 8
                                                       # pop two items from the stack
                             jalr
                                                       # return to the instruction after jal
                                        zero, ra, 0
                         L1: addi
                                        a0, a0, -1
                                                       # subtract 1 from the argument
                                        ra, fact
                             jal
                                                       # jump to fact and save position to ra
            0x0040 0040
                                        t0, a0
                                                       # copy the return value; don't overwrite
                             ΜV
                                        a0, 0(sp)
                                                       # just returned from jal, restore n
                             lw
                                        ra, 4(sp)
                                                       # restore the return address
                             lw
                                        sp, sp, 8
                             addi
                                                       # pop two items from the stack
            0x0040 0040
                                        a0, a0, t0
                                                       # return n * fact(n - 1)
                             mul
                             jalr
                                        zero, ra, 0
                                                       # return to the caller

    ↓ Low Address
```

Other Data Stored in the Stack



Recap

↑ High Address

Memory

个 High Address,	WEITIOLY	
, 6		
	0x0040	0008
	3	
	0x0040	0040
	2	
	0x0040	0040
	1	
	0x0040	0040
ا میں ۸ مامانیہ میا	0	
\downarrow Low Address		•