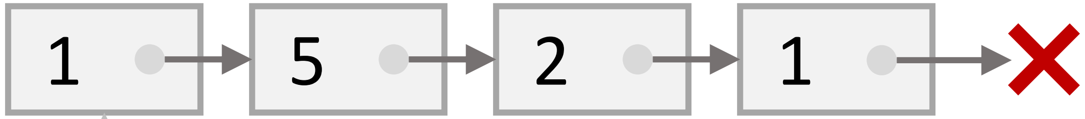
Recursion Examples

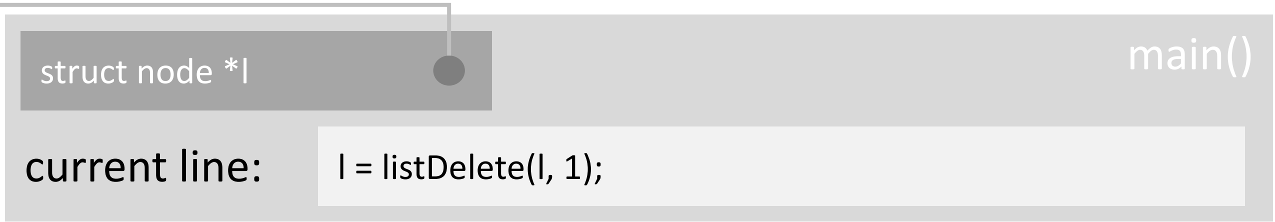
listDelete

Example 1

listDelete: Ex 1



```
int main(void) {
    ...
    l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

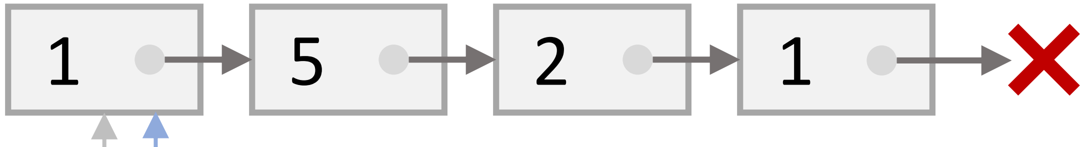


main()

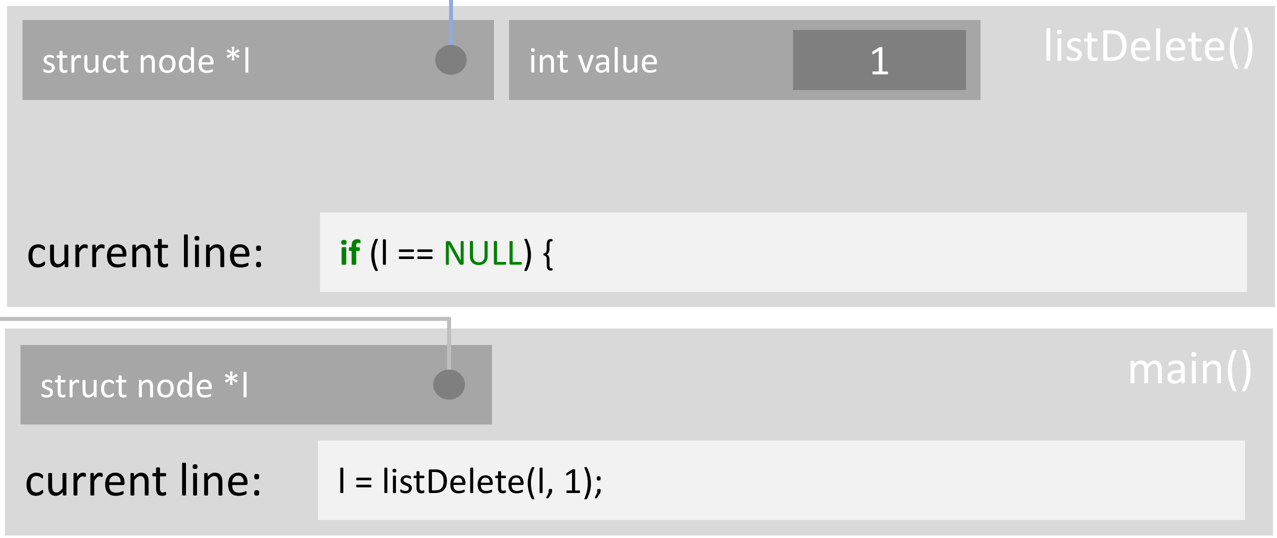
struct node *l

current line: l = listDelete(l, 1);

listDelete: Ex 1



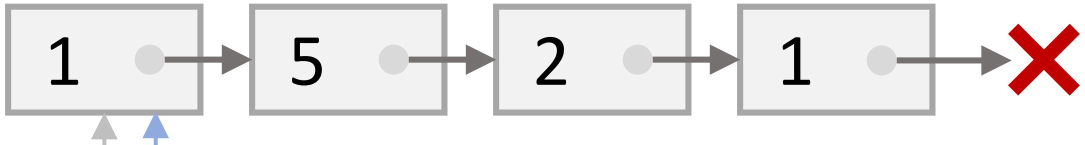
```
int main(void) {
   ...
   l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



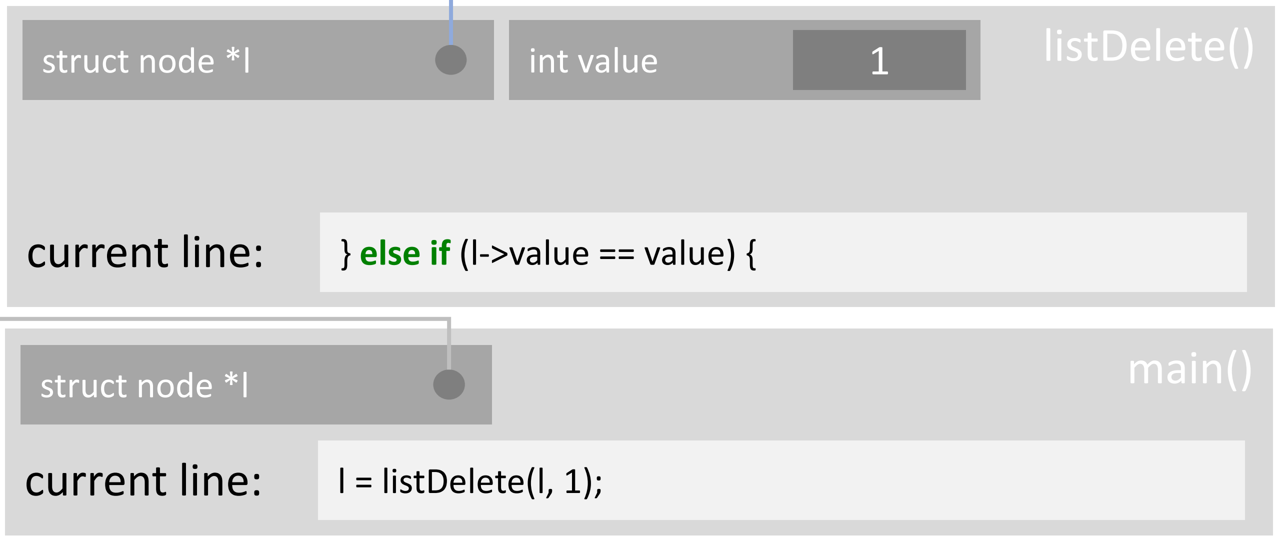
current line: `if (l == NULL) {`

current line: `l = listDelete(l, 1);`

listDelete: Ex 1



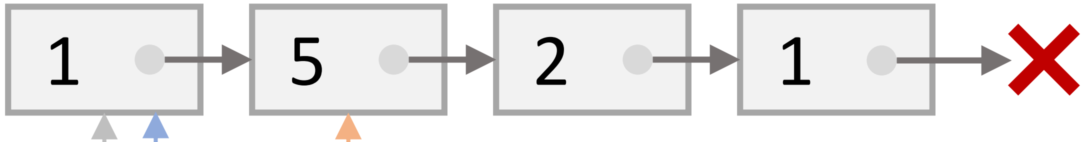
```
int main(void) {
    ...
    l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



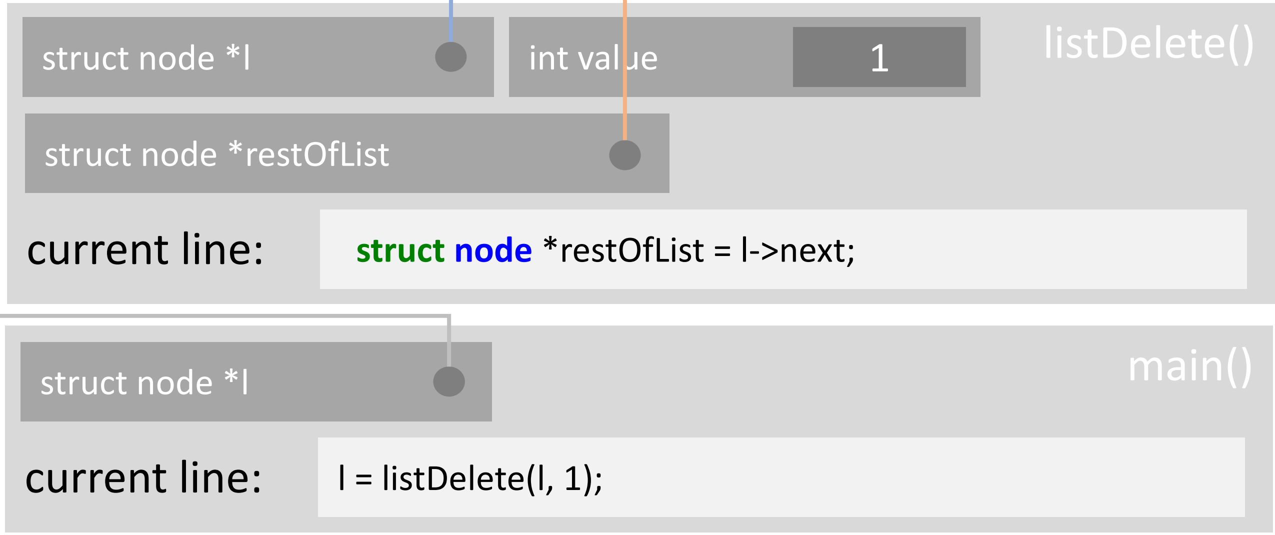
current line: } else if (l->value == value) {

current line: l = listDelete(l, 1);

listDelete: Ex 1



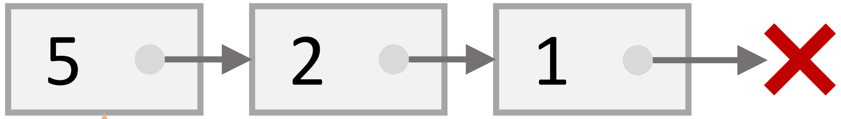
```
int main(void) {
   ...
   l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



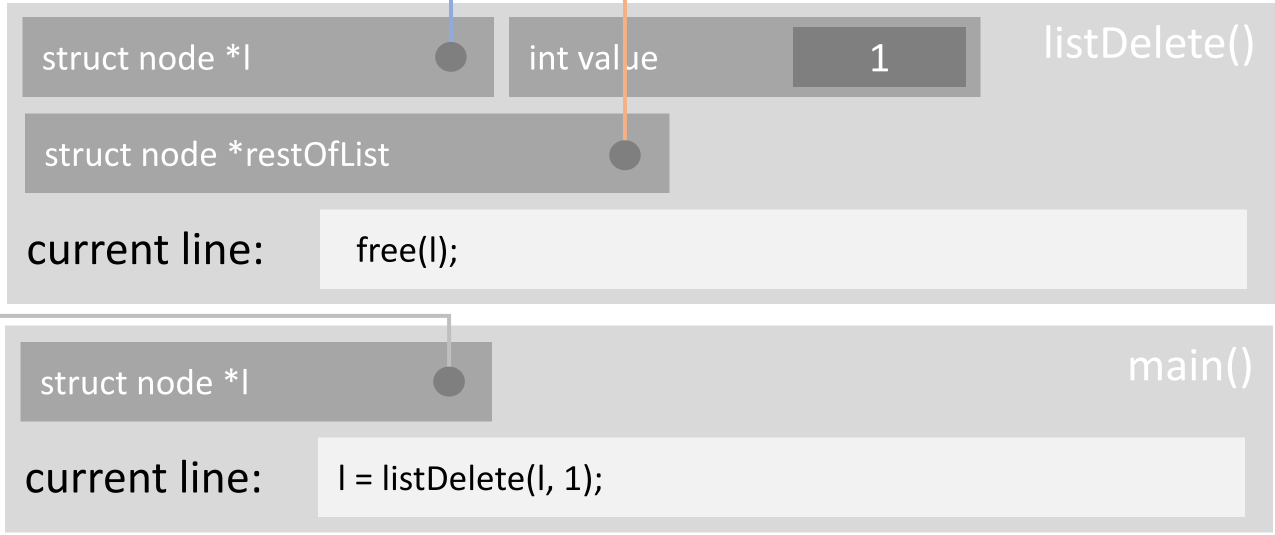
current line: `struct node *restOfList = l->next;`

current line: `l = listDelete(l, 1);`

listDelete: Ex 1



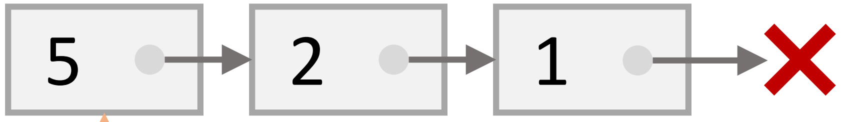
```
int main(void) {
    ...
    l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



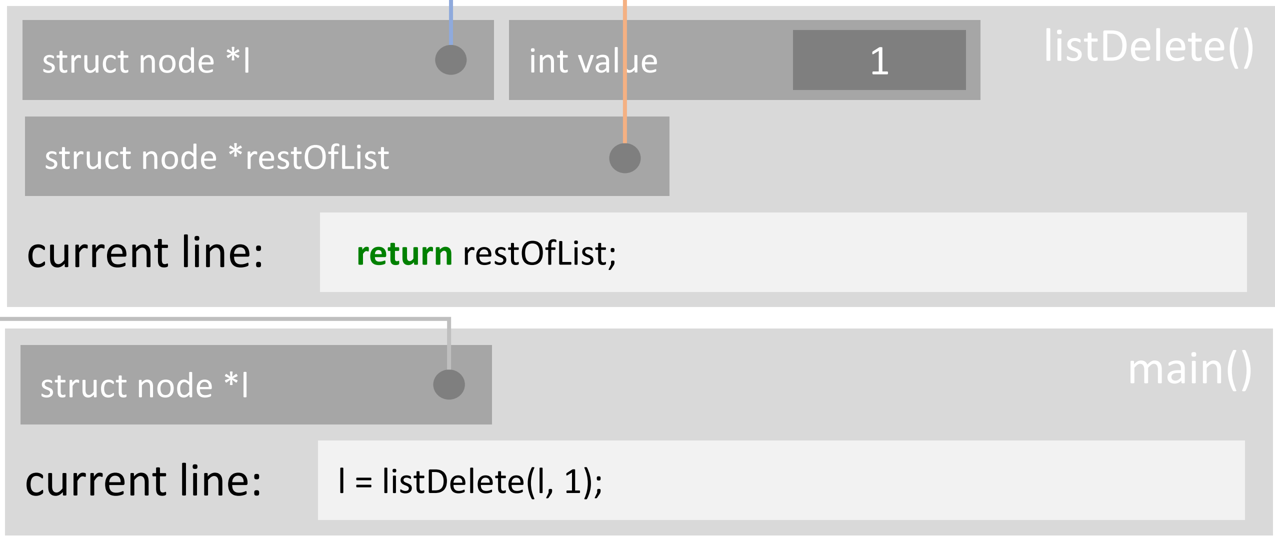
current line: free(l);

current line: l = listDelete(l, 1);

listDelete: Ex 1



```
int main(void) {
   ...
   l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



listDelete()

struct node *l

int value 1

struct node *restOfList

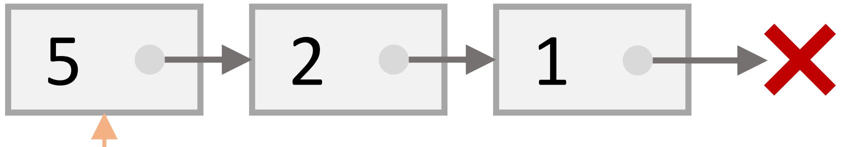
current line: **return** restOfList;

main()

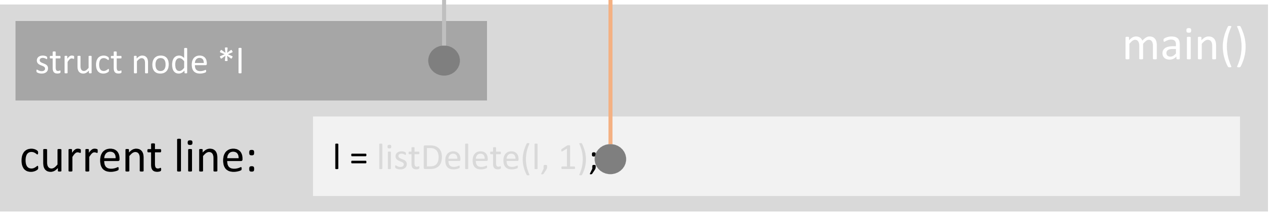
struct node *l

current line: l = listDelete(l, 1);

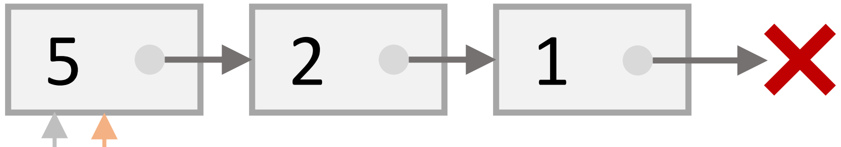
listDelete: Ex 1



```
int main(void) {
   ...
   l = listDelete(l, 1);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

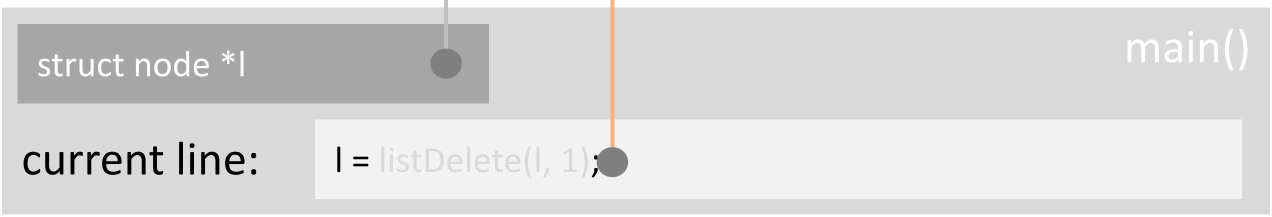


listDelete: Ex 1



```
int main(void) {
    ...
    l = listDelete(l, 1);
}
```

```
struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



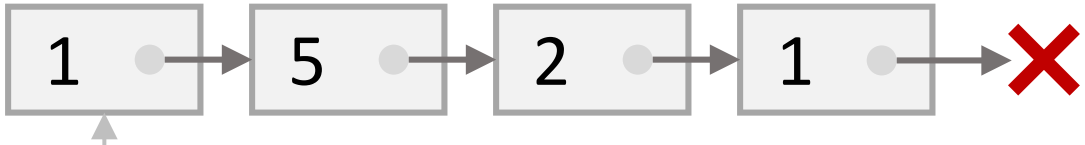
main()

struct node *l

current line: l = listDelete(l, 1);

Example 2

listDelete: Ex 2



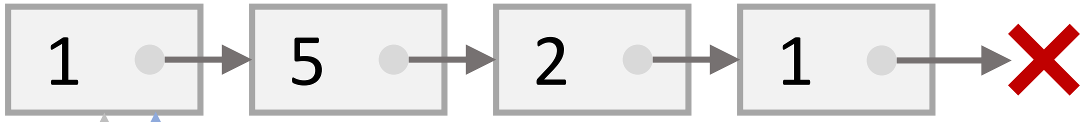
```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

main()

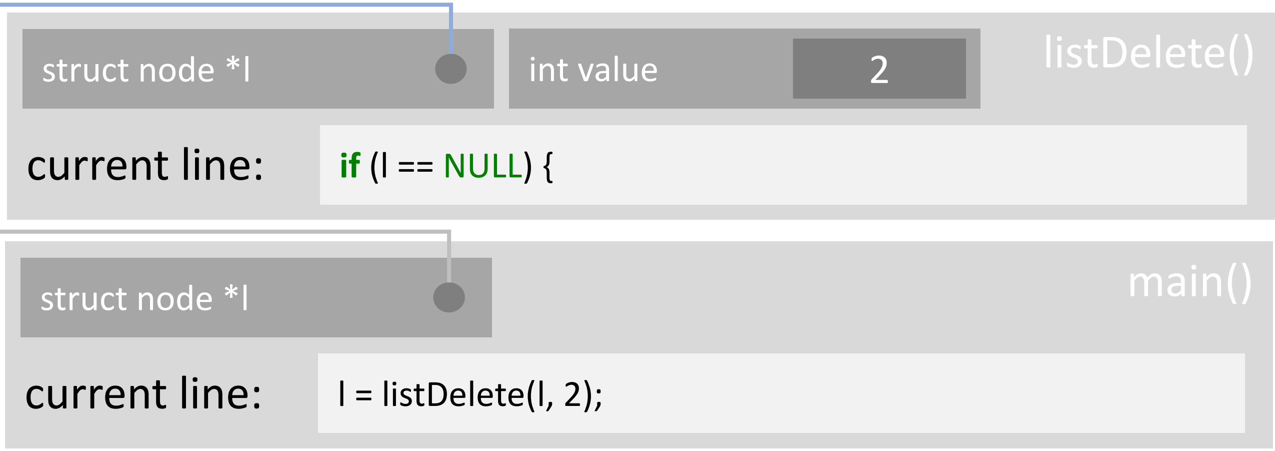
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete()

struct node *l

int value 2

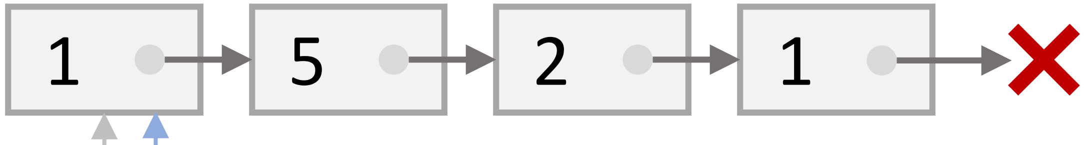
current line: if (l == NULL) {

main()

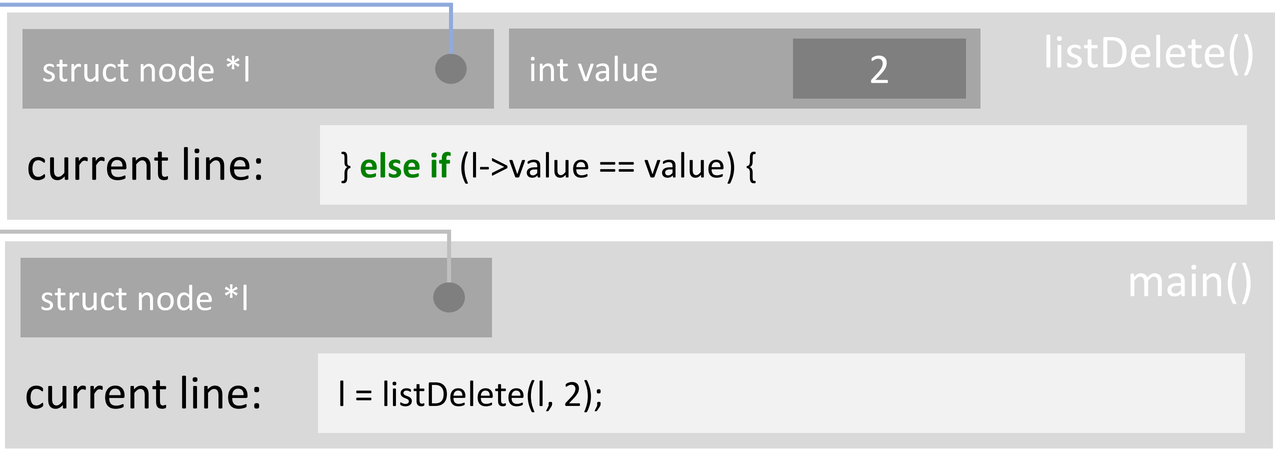
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete()

struct node *l

int value 2

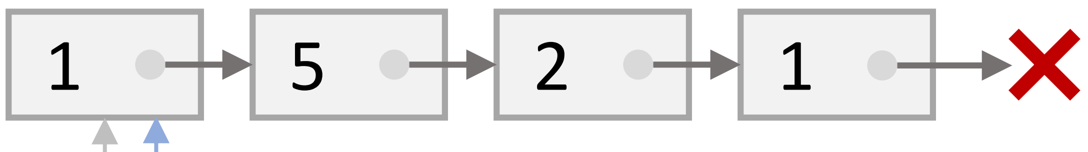
current line: } else if (l->value == value) {

main()

struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l

int value 2

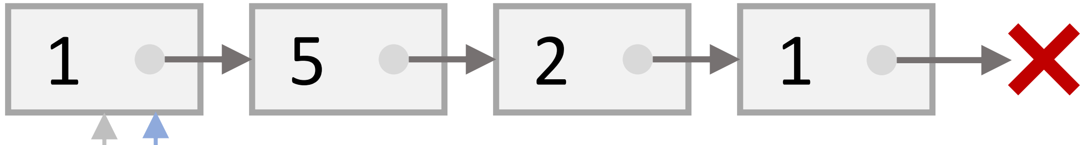
current line: } else {

main()

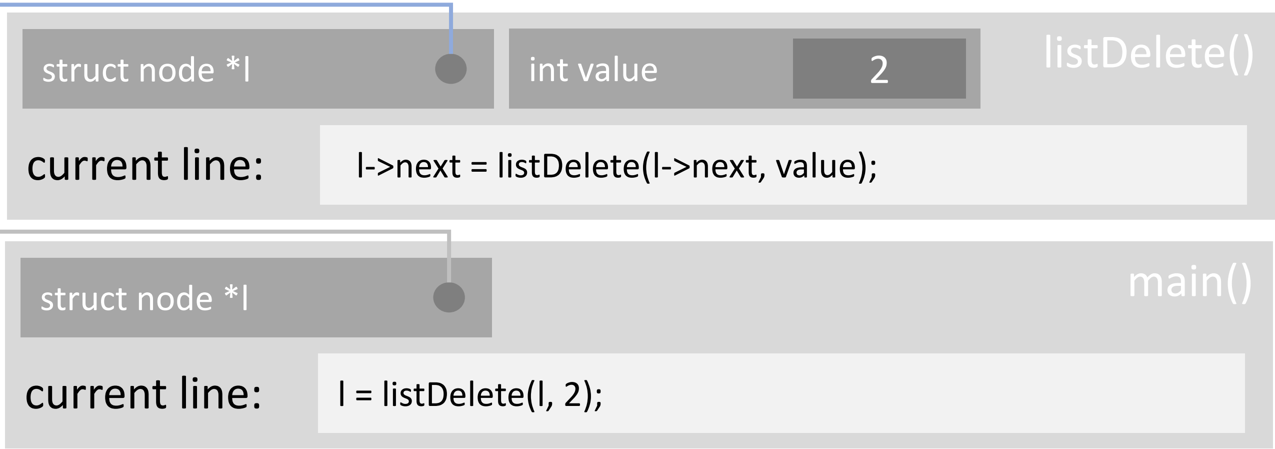
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



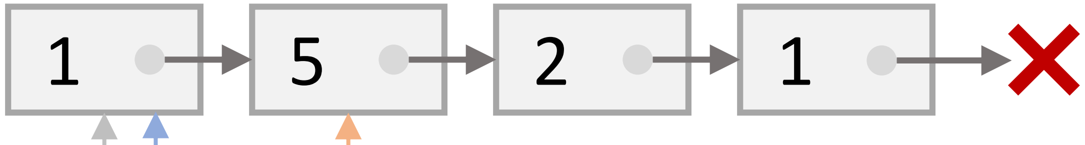
```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



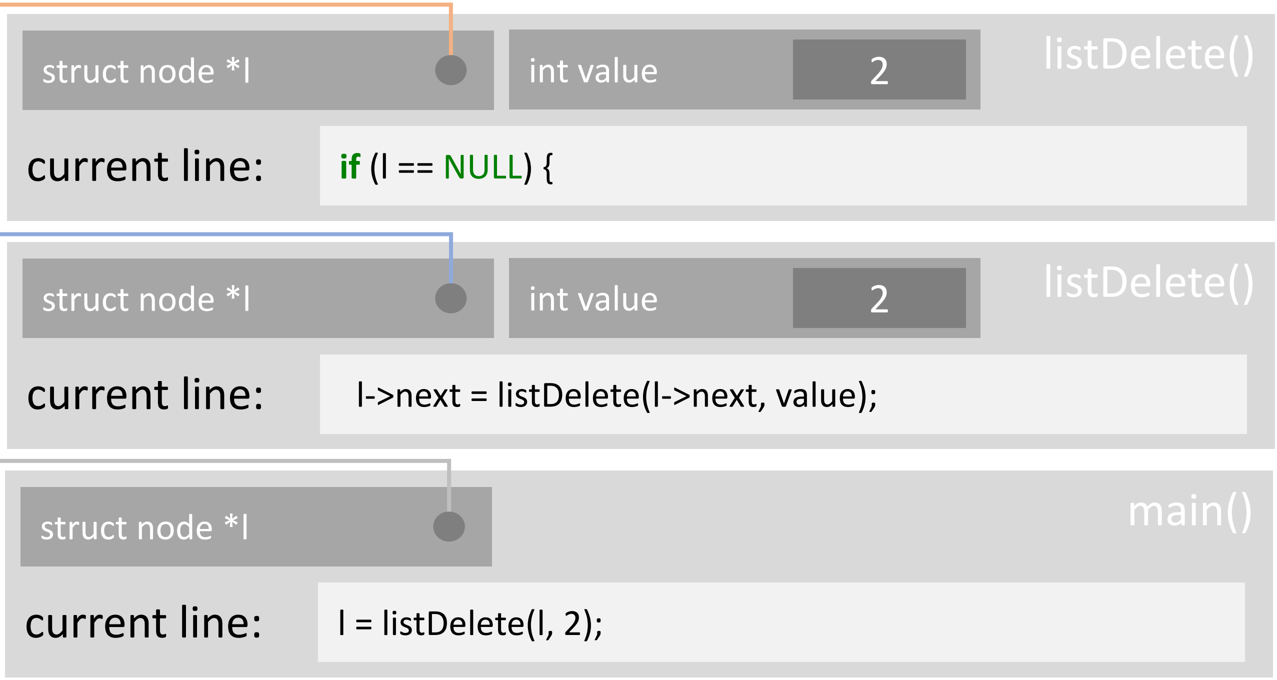
current line: l->next = listDelete(l->next, value);

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

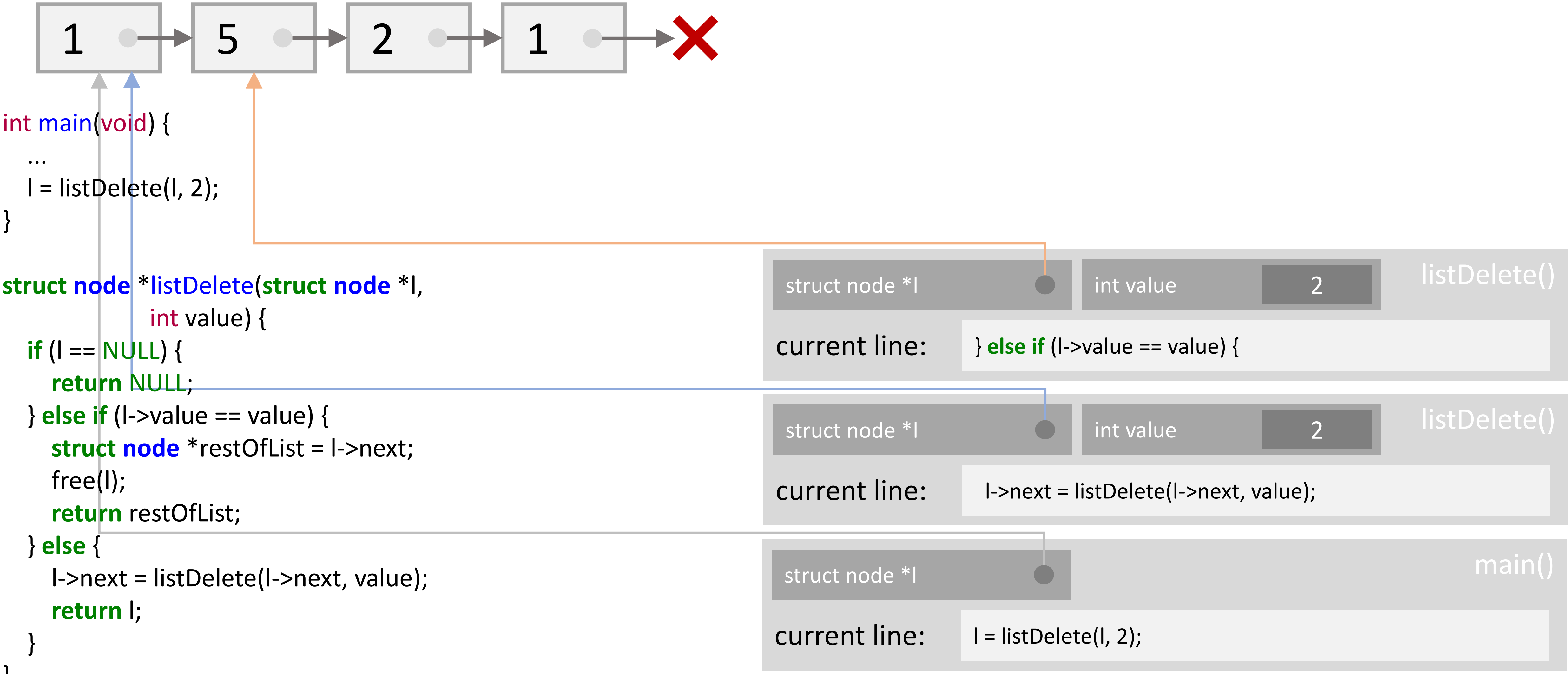


current line: `if (l == NULL) {`

current line: `l->next = listDelete(l->next, value);`

current line: `l = listDelete(l, 2);`

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

listDelete()

struct node *l

int value 2

current line: } else if (l->value == value) {

listDelete()

struct node *l

int value 2

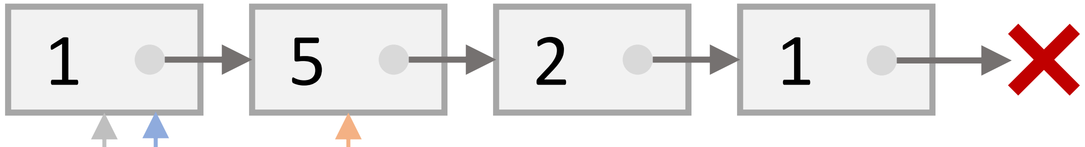
current line: l->next = listDelete(l->next, value);

main()

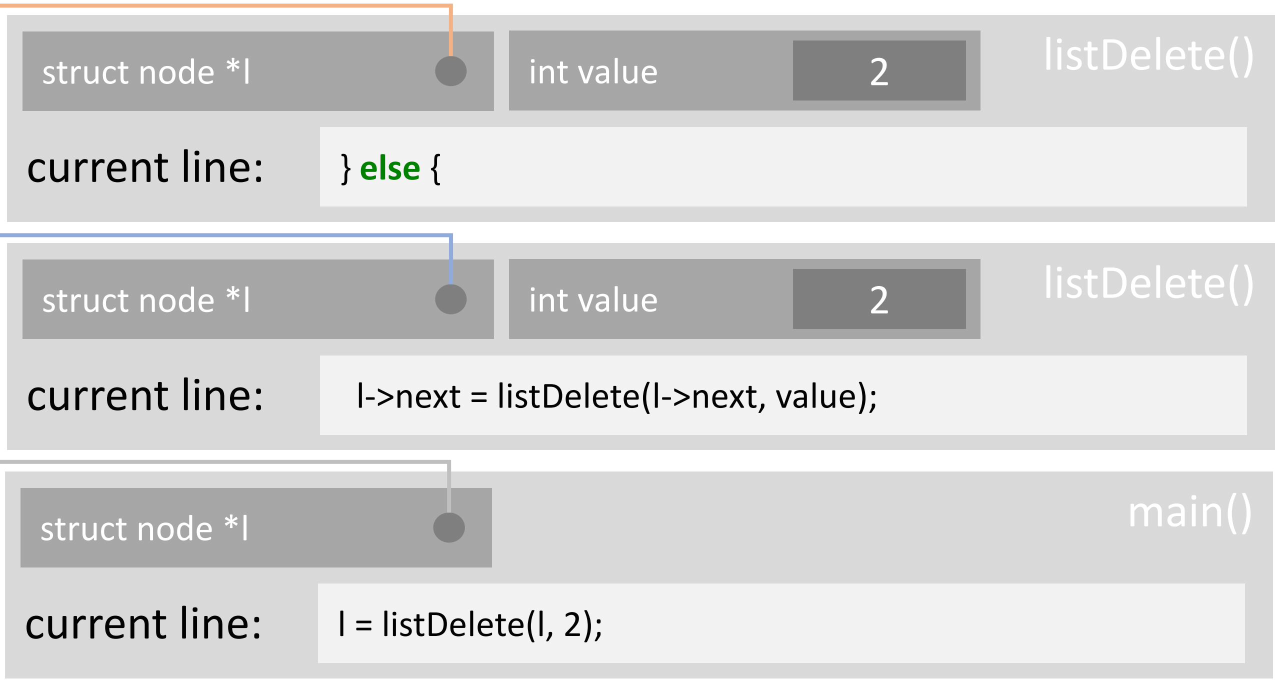
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete()

struct node *l

int value 2

current line: } else {

listDelete()

struct node *l

int value 2

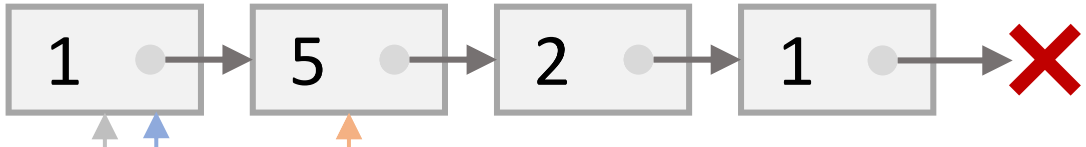
current line: l->next = listDelete(l->next, value);

main()

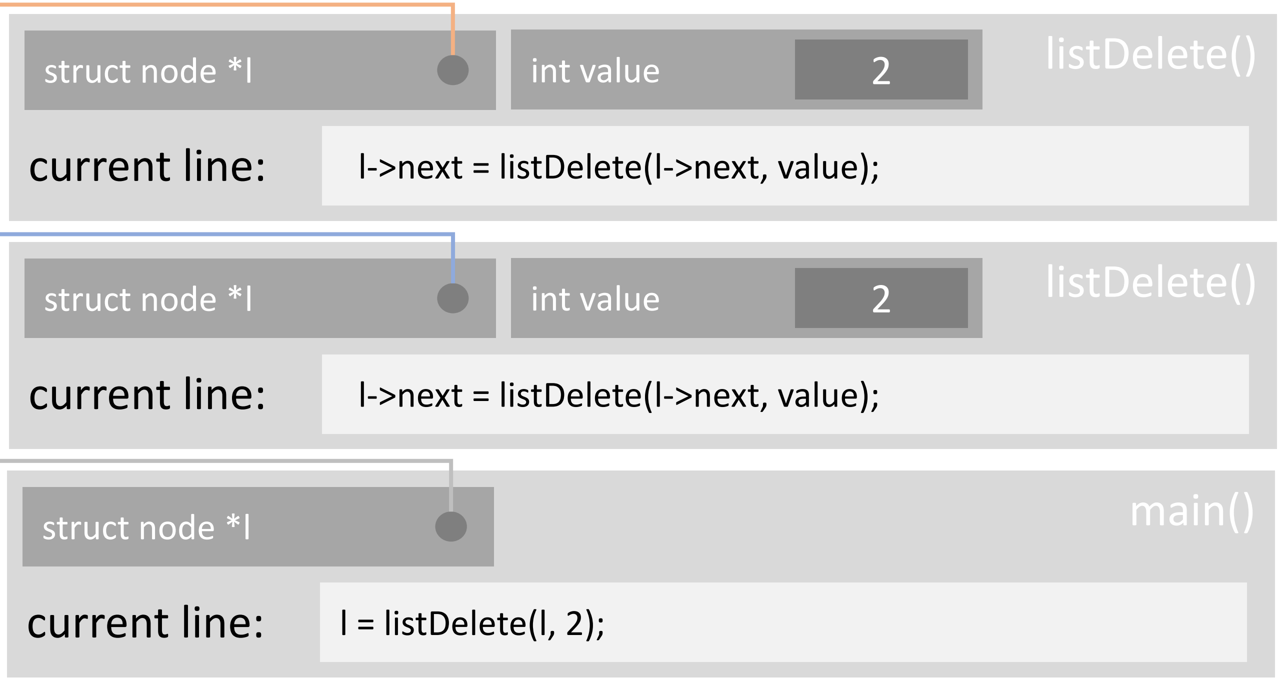
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

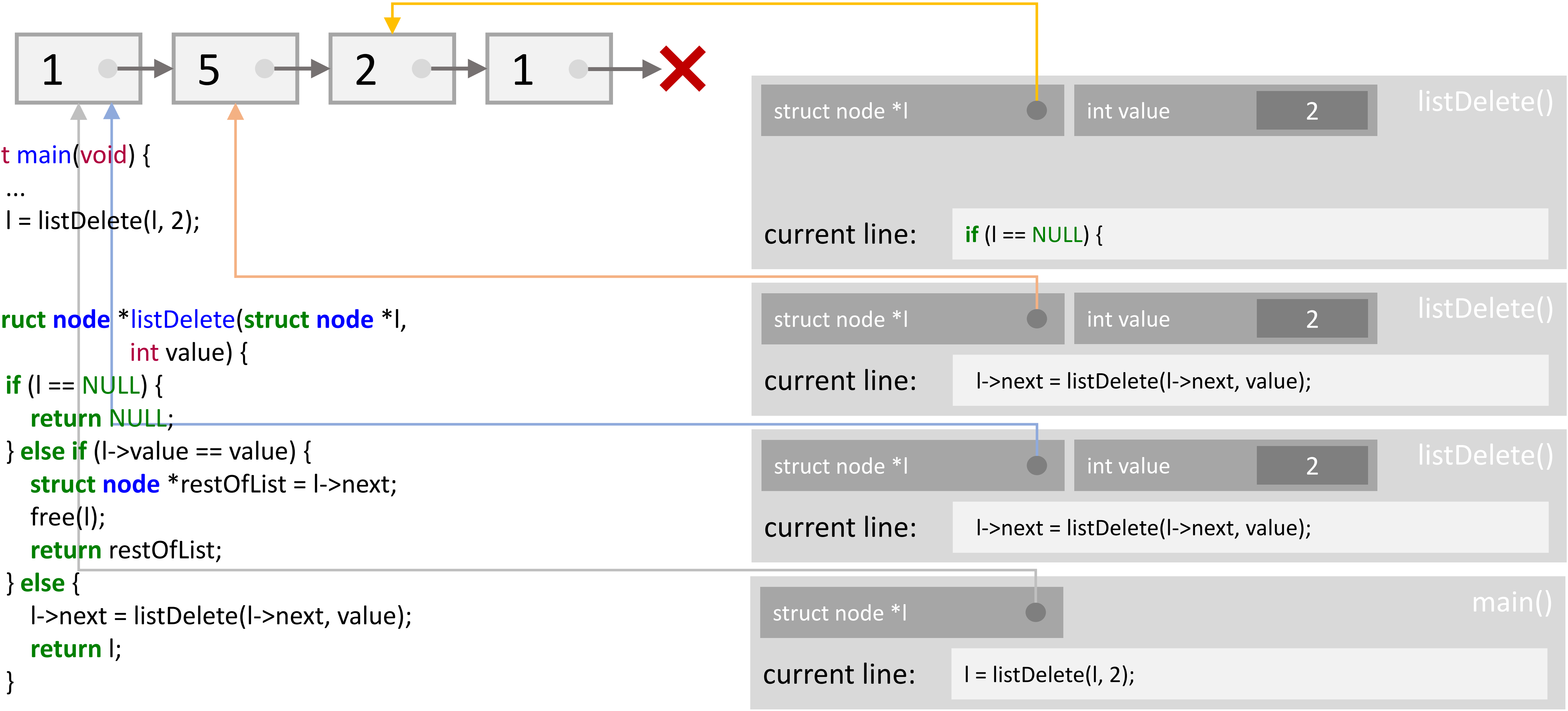


l->next = listDelete(l->next, value);

l->next = listDelete(l->next, value);

l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l | int value 2

current line: `if (l == NULL) {`

listDelete()

struct node *l | int value 2

current line: `l->next = listDelete(l->next, value);`

listDelete()

struct node *l | int value 2

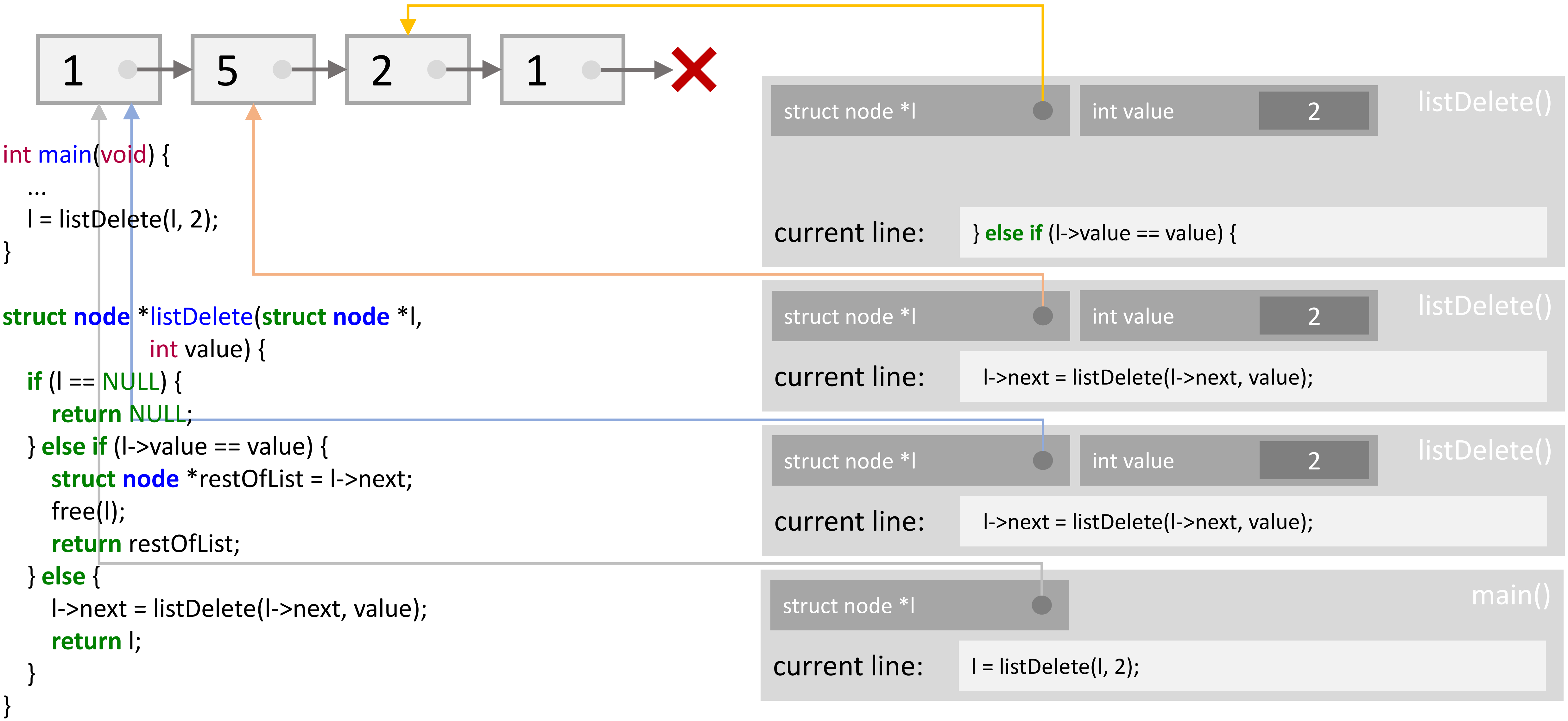
current line: `l->next = listDelete(l->next, value);`

main()

struct node *l

current line: `l = listDelete(l, 2);`

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l

int value 2

current line: `} else if (l->value == value) {`

listDelete()

struct node *l

int value 2

current line: `l->next = listDelete(l->next, value);`

listDelete()

struct node *l

int value 2

current line: `l->next = listDelete(l->next, value);`

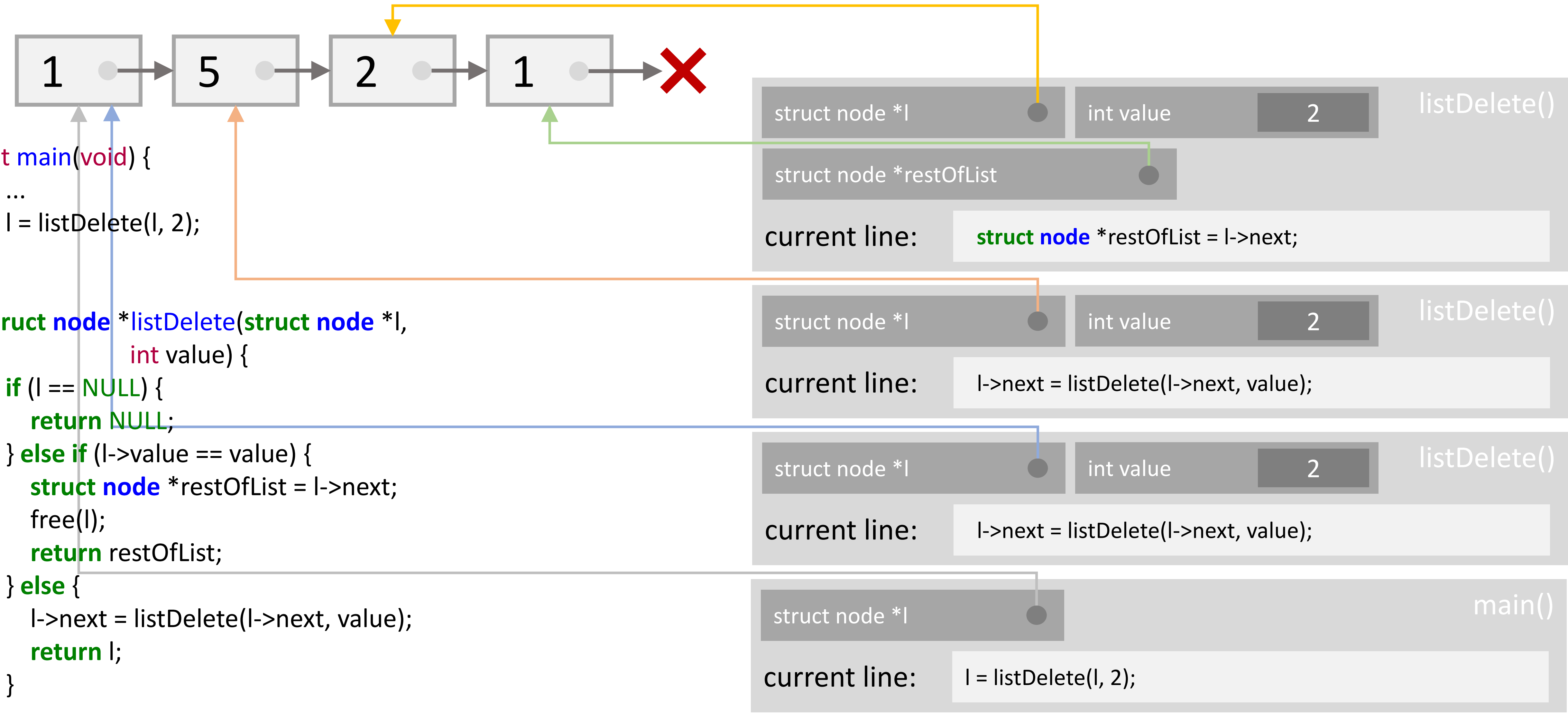
main()

struct node *l

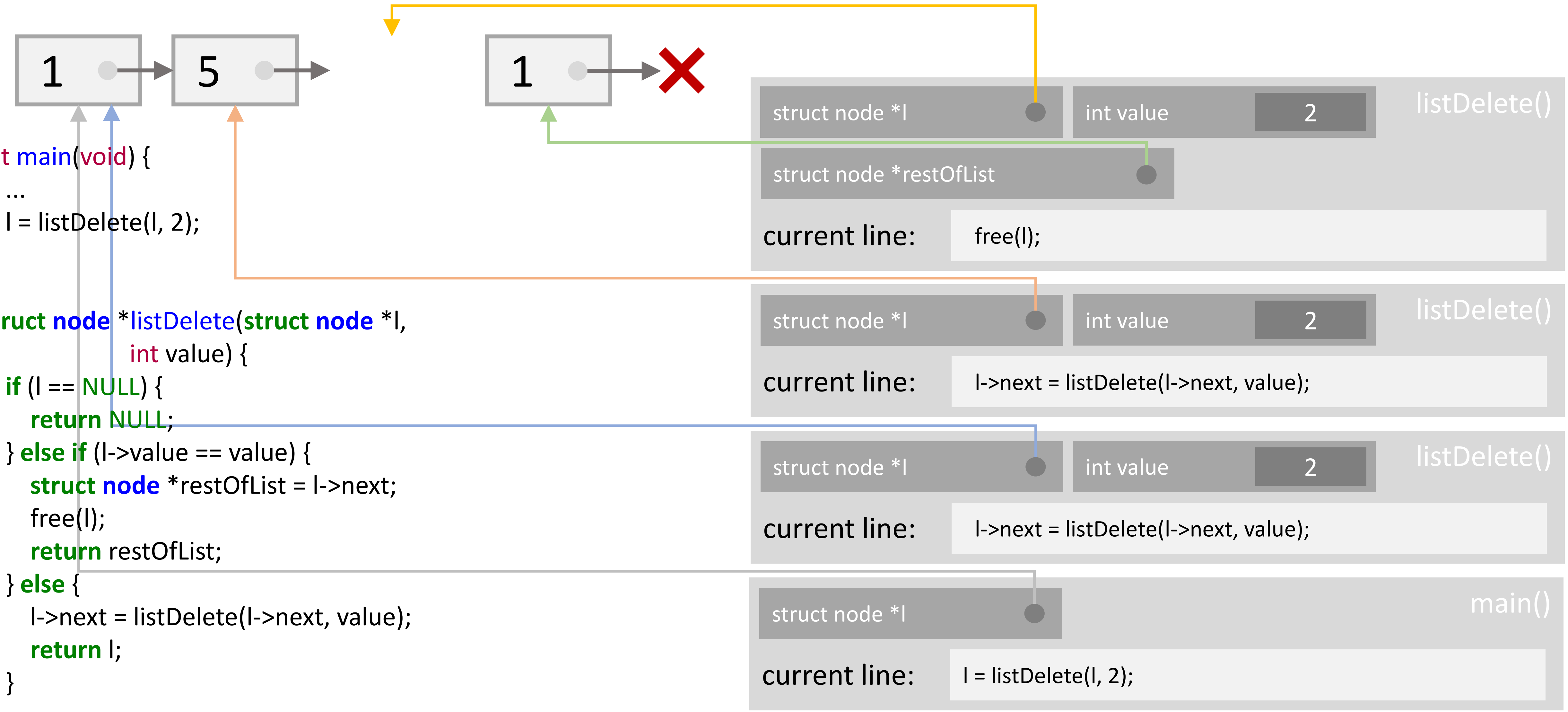
current line: `l = listDelete(l, 2);`

listDelete: Ex 2

```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l	int value 2
struct node *restOfList	
current line:	free(l);

listDelete()

struct node *l	int value 2
current line:	l->next = listDelete(l->next, value);

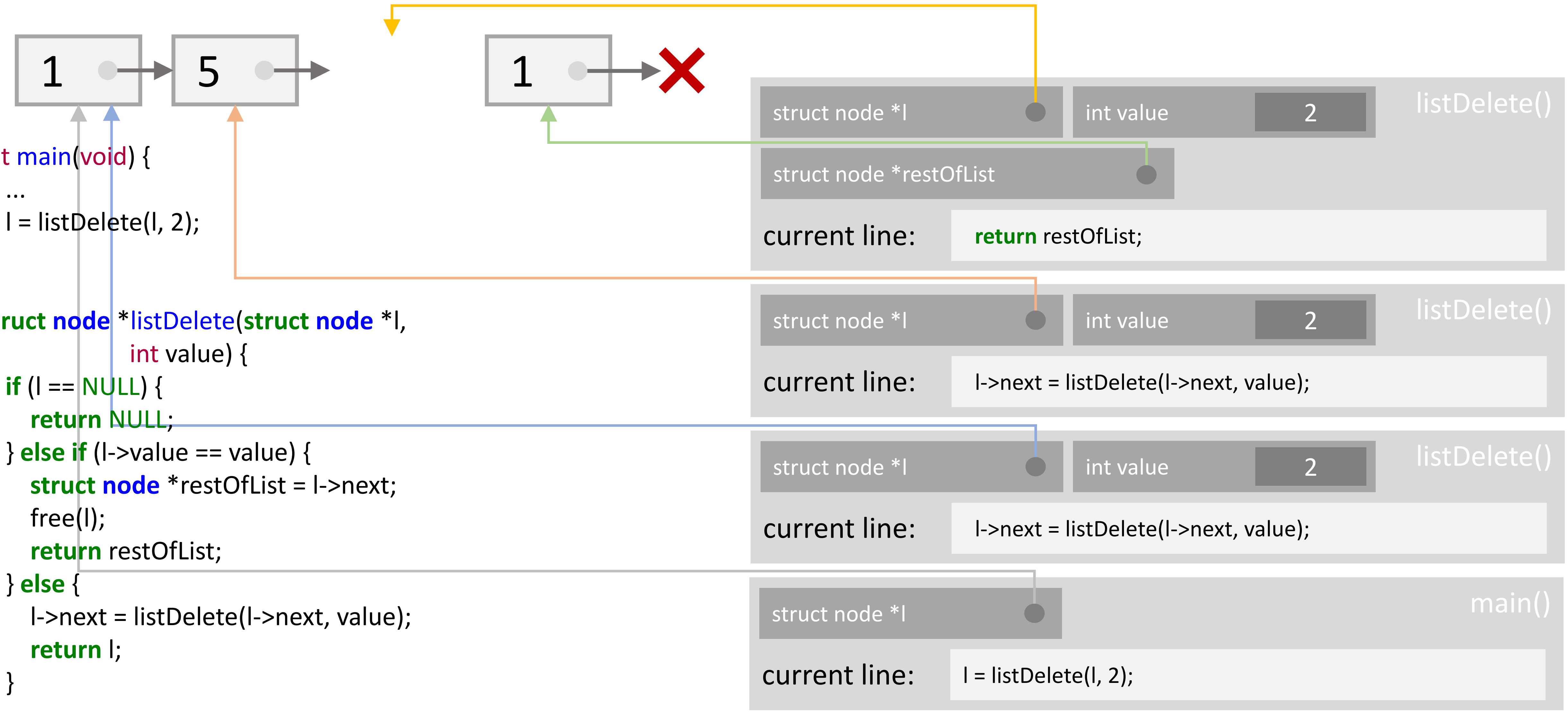
listDelete()

struct node *l	int value 2
current line:	l->next = listDelete(l->next, value);

main()

struct node *l	
current line:	l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

listDelete()

struct node *l

int value 2

struct node *restOfList

current line: `return restOfList;`

listDelete()

struct node *l

int value 2

current line: `l->next = listDelete(l->next, value);`

listDelete()

struct node *l

int value 2

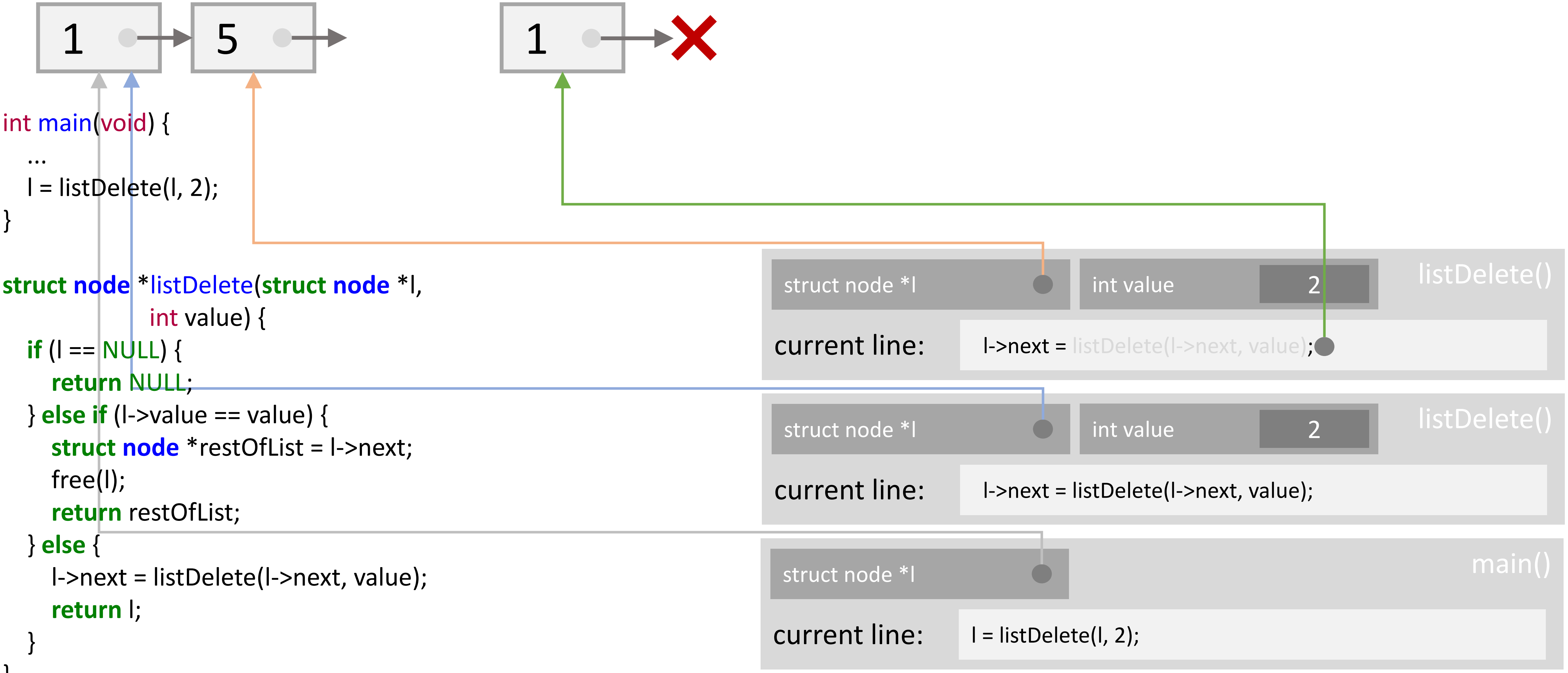
current line: `l->next = listDelete(l->next, value);`

main()

struct node *l

current line: `l = listDelete(l, 2);`

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```

listDelete()

struct node *l

int value 2

current line: l->next = listDelete(l->next, value);

listDelete()

struct node *l

int value 2

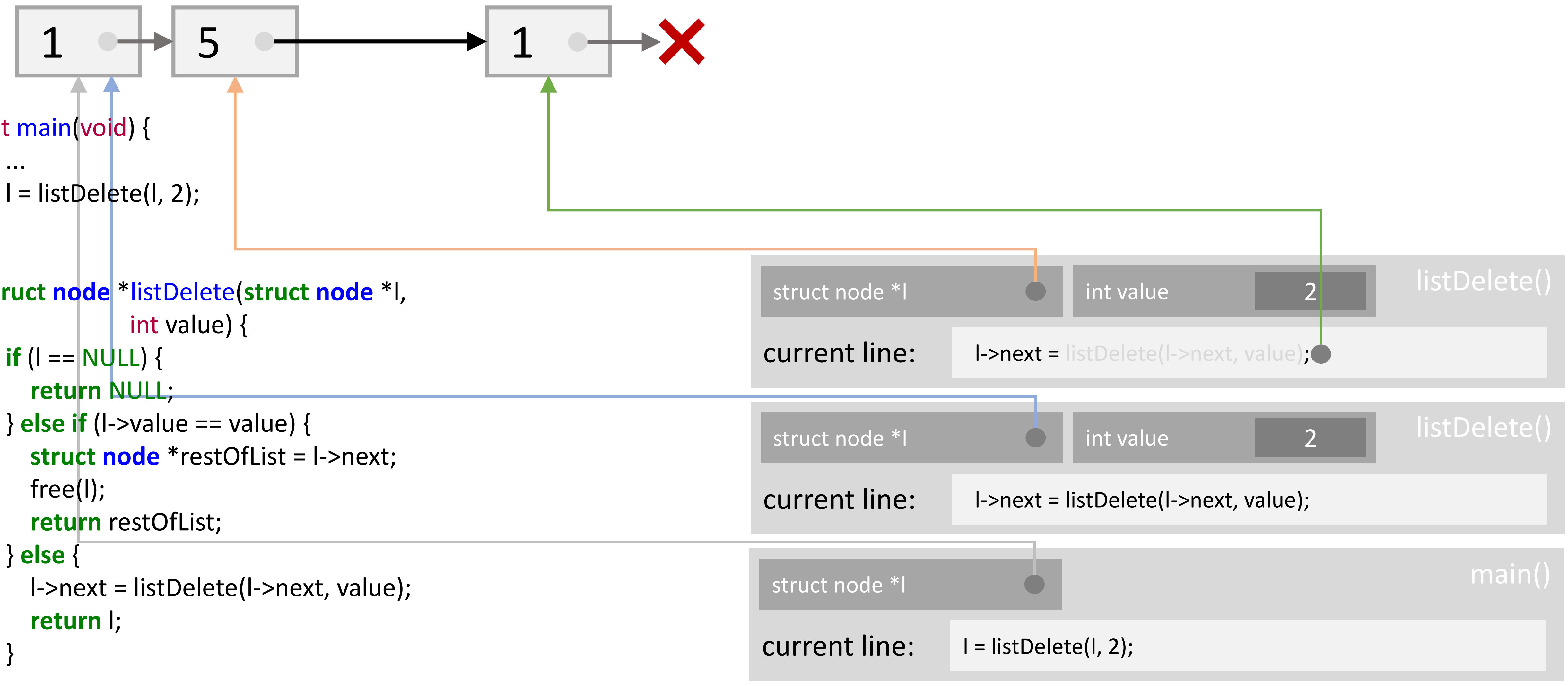
current line: l->next = listDelete(l->next, value);

main()

struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l

int value 2

current line: l->next = listDelete(l->next, value);

listDelete()

struct node *l

int value 2

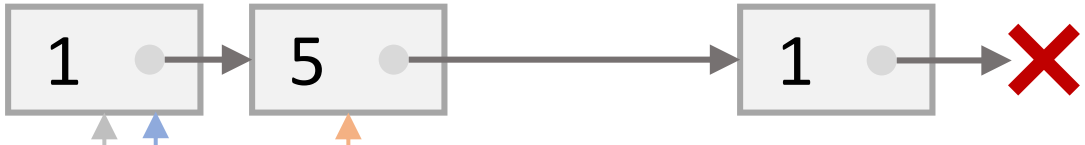
current line: l->next = listDelete(l->next, value);

main()

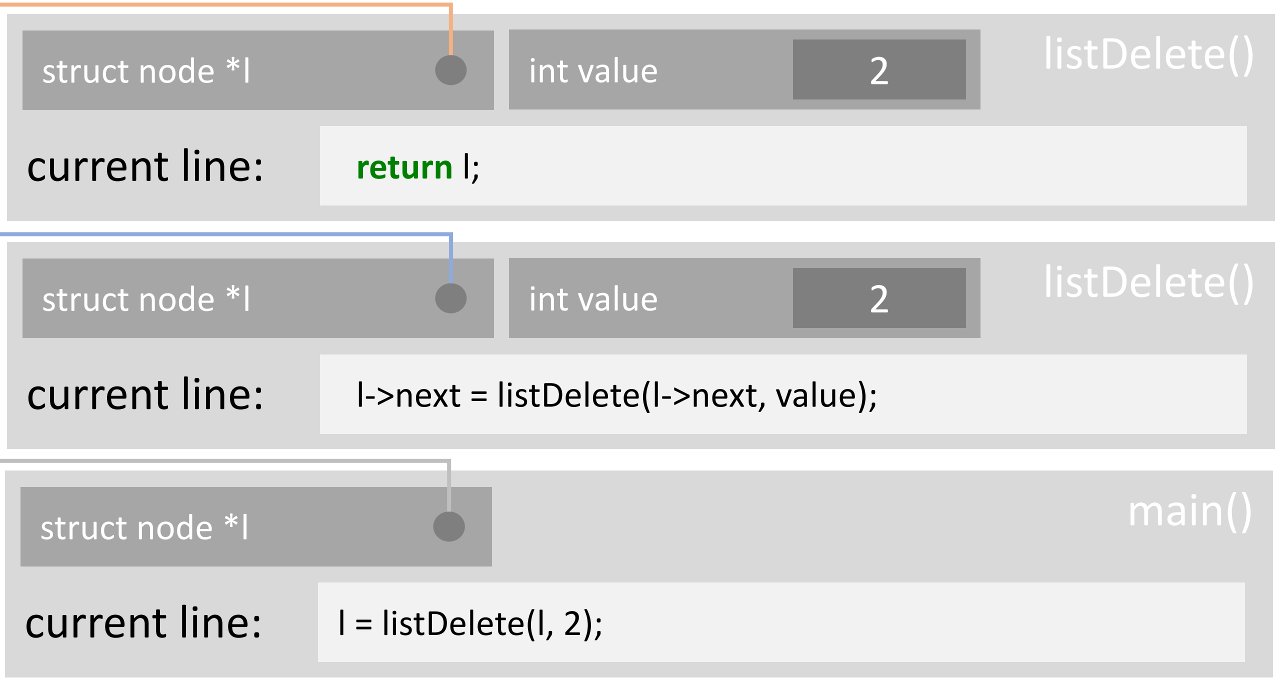
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```



listDelete()

struct node *l | int value 2

current line: return l;

listDelete()

struct node *l | int value 2

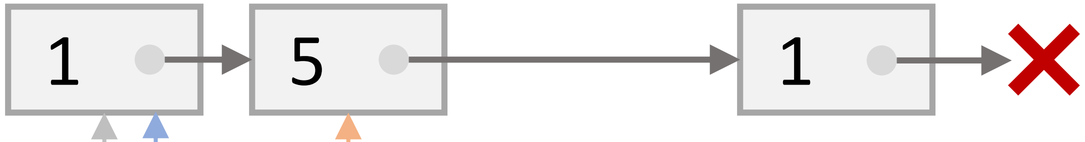
current line: l->next = listDelete(l->next, value);

main()

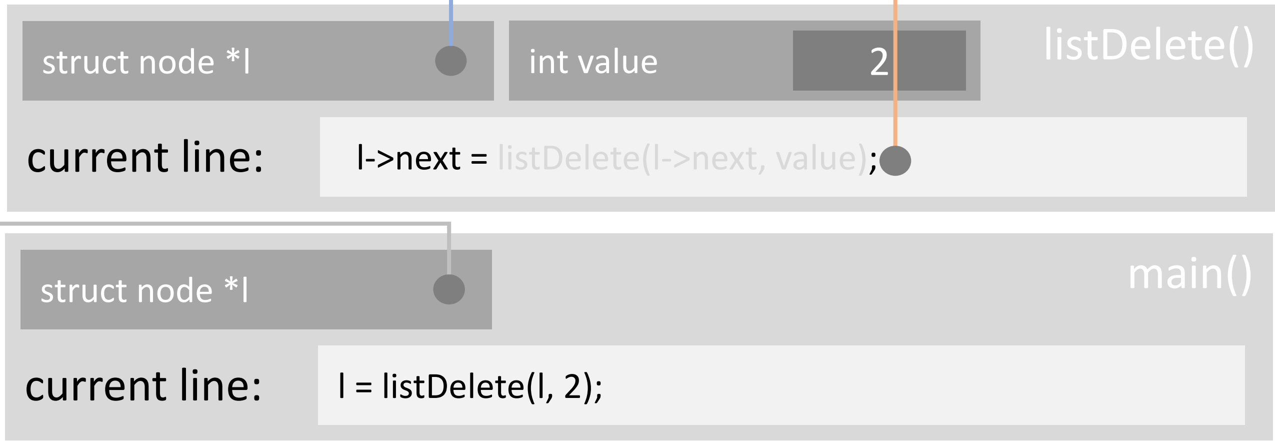
struct node *l

current line: l = listDelete(l, 2);

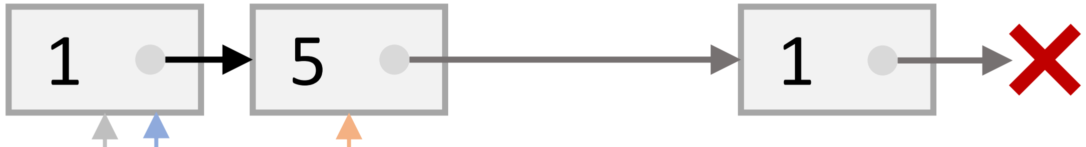
listDelete: Ex 2



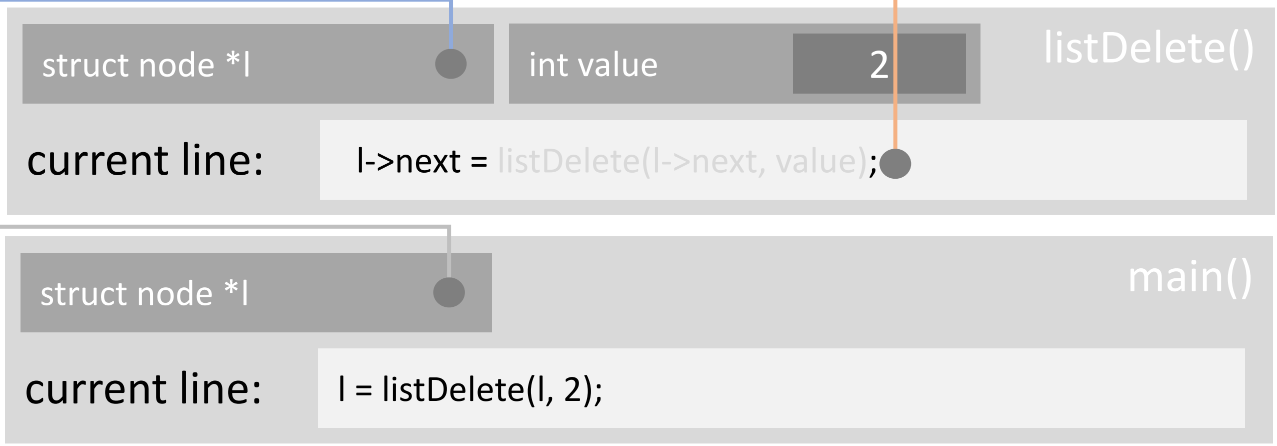
```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete()

struct node *l	int value	2

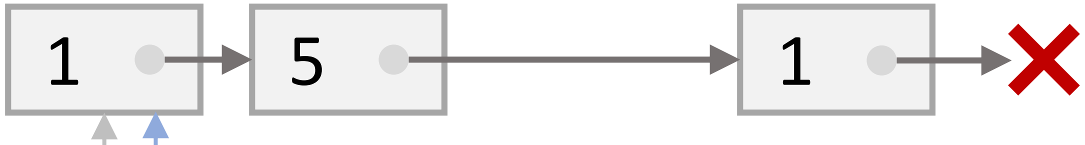
current line: l->next = listDelete(l->next, value);

main()

struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

listDelete()

struct node *l ●

int value 2

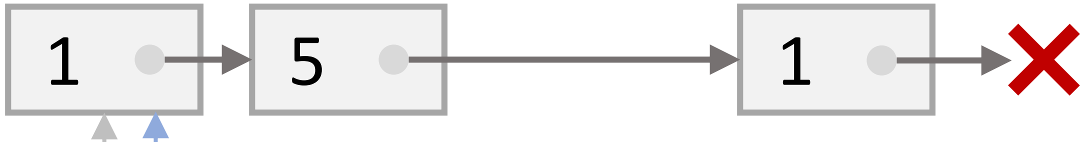
current line: return l;

main()

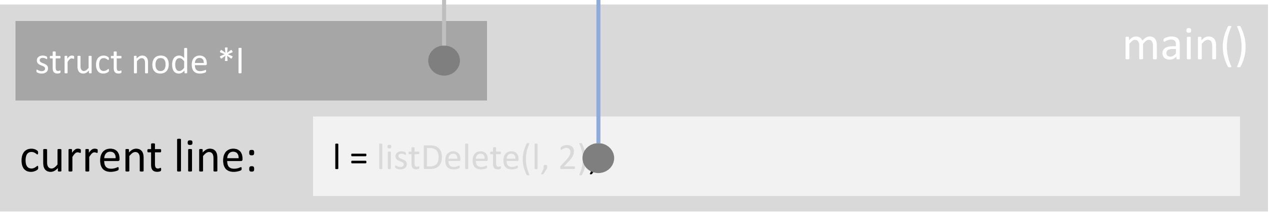
struct node *l ●

current line: l = listDelete(l, 2);

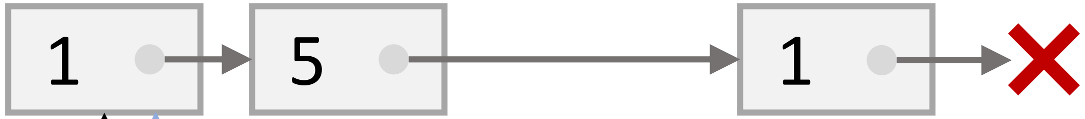
listDelete: Ex 2



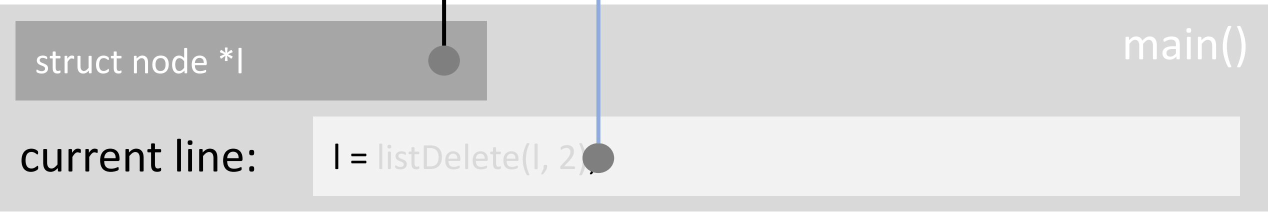
```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```



listDelete: Ex 2



```
int main(void) {
    ...
    l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
    if (l == NULL) {
        return NULL;
    } else if (l->value == value) {
        struct node *restOfList = l->next;
        free(l);
        return restOfList;
    } else {
        l->next = listDelete(l->next, value);
        return l;
    }
}
```

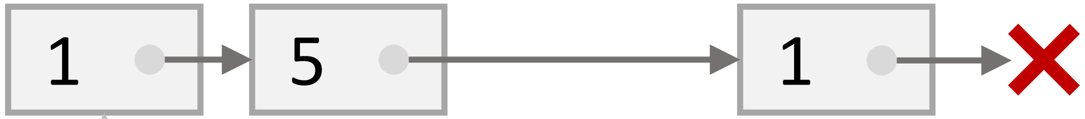


main()

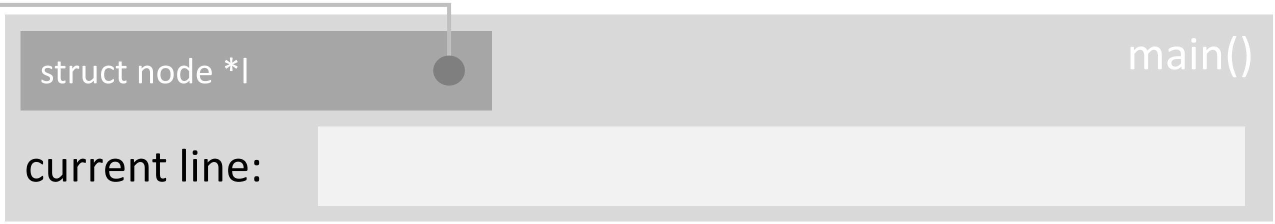
struct node *l

current line: l = listDelete(l, 2);

listDelete: Ex 2



```
int main(void) {
   ...
   l = listDelete(l, 2);
}

struct node *listDelete(struct node *l,
                        int value) {
   if (l == NULL) {
      return NULL;
   } else if (l->value == value) {
      struct node *restOfList = l->next;
      free(l);
      return restOfList;
   } else {
      l->next = listDelete(l->next, value);
      return l;
   }
}
```


struct node *l
main()
current line: