
Article wrapper

1. "Default" rendering

A funcsynopsis.

```
<funcsynopsis>
<funcsynopsisinfo> #include <varargs.h> </funcsynopsisinfo>
<funcprototype>
<funcdef>float rand</funcdef>
<void></void>
</funcprototype>
<funcprototype>
<funcdef>int max</funcdef>
<varargs></varargs>
</funcprototype>
<funcprototype>
<funcdef>int idiv</funcdef>
<paramdef>int n</paramdef>
<paramdef>int m</paramdef>
</funcprototype>
</funcsynopsis>
```

Another.

```
<funcsynopsis>
<funcprototype>
<funcdef>void qsort</funcdef>
<paramdef>void *dataptr[]</paramdef>
<paramdef>int left</paramdef>
<paramdef>int right</paramdef>
<paramdef>int (*comp)
<funcparams>void *, void *</funcparams>
</paramdef>
</funcprototype>
</funcsynopsis>
```

Another.

```
<funcsynopsis>
<funcprototype>
<funcdef>int foo_frob_something</funcdef>
<paramdef>foo_sometypel foo_frob_parm1</paramdef>
<paramdef>foo_sometypel foo_frob_parm2</paramdef>
<paramdef>foo_sometypel foo_frob_parm3</paramdef>
<paramdef>int (* parm4 )
<funcparams>int a, int b, int c</funcparams>
</paramdef>
<paramdef>foo_sometypel foo_frob_parm5</paramdef>
</funcprototype>
<funcprototype>
<funcdef>int foo_frob_something</funcdef>
<paramdef>foo_sometypel foo_frob_parm1</paramdef>
<paramdef>foo_sometypel foo_frob_parm2</paramdef>
```

```

<paramdef>foo_sometype1 foo_frob_parm3</paramdef>
<paramdef>int (* parm4 )
<funcparams>int a, int b, int c</funcparams>
</paramdef>
<paramdef>foo_sometype1 foo_frob_parm5</paramdef>
</funcprototype>
</funcsynopsis>

```

2. ANSI Rendering

Another.

```

<funcsynopsis>
<funcsynopsisinfo> #include <varargs.h> </funcsynopsisinfo>
<funcprototype>
<funcdef>float rand</funcdef>
<void></void>
</funcprototype>
<funcprototype>
<funcdef>int max</funcdef>
<varargs></varargs>
</funcprototype>
<funcprototype>
<funcdef>int idiv</funcdef>
<paramdef>int n</paramdef>
<paramdef>int m</paramdef>
</funcprototype>
</funcsynopsis>

```

Another.

```

<funcsynopsis>
<funcprototype>
<funcdef>void qsort</funcdef>
<paramdef>void *dataptr[]</paramdef>
<paramdef>int left</paramdef>
<paramdef>int right</paramdef>
<paramdef>int (*comp)
<funcparams>void *, void *</funcparams>
</paramdef>
</funcprototype>
</funcsynopsis>

```

Another.

```

<funcsynopsis>
<funcprototype>
<funcdef>int foo_frob_something</funcdef>
<paramdef>foo_sometype1 foo_frob_parm1</paramdef>
<paramdef>foo_sometype1 foo_frob_parm2</paramdef>
<paramdef>foo_sometype1 foo_frob_parm3</paramdef>
<paramdef>int (* parm4 )
<funcparams>int a, int b, int c</funcparams>
</paramdef>
<paramdef>foo_sometype1 foo_frob_parm5</paramdef>

```

```
</funcprototype>
</funcsynopsis>
```

3. K&R Rendering

Another.

```
<funcsynopsis>
<funcsynopsisinfo> #include <varargs.h> </funcsynopsisinfo>
<funcprototype>
<funcdef>float rand</funcdef>
<void></void>
</funcprototype>
<funcprototype>
<funcdef>int max</funcdef>
<varargs></varargs>
</funcprototype>
<funcprototype>
<funcdef>int idiv</funcdef>
<paramdef>int n</paramdef>
<paramdef>int m</paramdef>
</funcprototype>
</funcsynopsis>
```

Another.

```
<funcsynopsis>
<funcprototype>
<funcdef>void qsort</funcdef>
<paramdef>void *dataptr[]</paramdef>
<paramdef>int left</paramdef>
<paramdef>int right</paramdef>
<paramdef>int (*comp)
<funcparams>void *, void *</funcparams>
</paramdef>
</funcprototype>
</funcsynopsis>
```

Another.

```
<funcsynopsis>
<funcprototype>
<funcdef>int foo_frob_something</funcdef>
<paramdef>foo_sometype1 foo_frob_parm1</paramdef>
<paramdef>foo_sometype1 foo_frob_parm2</paramdef>
<paramdef>foo_sometype1 foo_frob_parm3</paramdef>
<paramdef>int (* parm4 )
<funcparams>int a, int b, int c</funcparams>
</paramdef>
<paramdef>foo_sometype1 foo_frob_parm5</paramdef>
</funcprototype>
</funcsynopsis>
```