

There are two approaches you can take to use this document: the first is to look over the table of work to be completed and find something you are interested in, then use the rank listed and subsequent description to find out how to acquire the skill set necessary to complete that task.

The second is to look over the description of ranks, decide what ranks will be easiest for you to attain, and then look over the table of work for which tasks you would then be capable of completing.

#### Problems\Annoyances noted during IGVC2009:

symptom	severity	probable cause
loss of control at maximum speed	annoyance	unknown
undesirable behavior during GPS nav	critical	unknown
gets trapped in corners	serious	obs avoid algorithm
periodic jerky movements (see MVI_2201.avi)	serious	Multiple, including matlab image processing latency?
Info extracted from cameras wrong	serious	vision alg calibrated for narrow range of conditions

The ability levels in the table are expressed as ranks, which are described following this table. The abbreviations should become obvious after glancing at that page. The priority levels mean approximately:

Priority 1: must be done; not optional

2: will directly and automatically improve score at next competition

3: should be done by next competition, benefit hard to quantify

4: longer term projects that offer potential of significant performance increase

5: should be done out of principle, but probably won't effect anything

6: purely for fun

2009-2010 planned improvements	priority	Apprx ability level
Debug& fix GPS	1	HI2&O2
Complete design report for IGVC 2010	1	O2&HI2&A2&M2
Complete presentation for IGVC 2010	1	O2&HI2&A2&M2
do paperwork to get 2010-2011 year funding	1	ST3
Implement JAUS lib	2	A1&HI2
Convert vision processing to opencv	2	HI3
Implement new\modified obs avoid algorithm	2	A3
Covering\waterproofing	3	M1
Implement source control for lab	3	A0

Improve vision algorithm	3	A0-A3
Integrate stereovision	4	H13
Use second laser for backing up	4	H13&A2
Wide angle lens hardware&algorithm	4	A2&H12
Mapping\SLAM	4	A3
Implement RS422 for laser	5	H12
Debug speed control	5	H12&O2
Fix old laptop firewire port	5	H2-3
integrating microphones into code	5	H13&A2
Streaming media server	6	A1
Add microbots to cub	6	A1&M1
Network based remote ops	6	A1&H11

## Bearcat Cub Micro Ranks

I hope that being able to offer people promotions along the different tracks will provide some additional motivation for learning. If necessary, I may produce badges which people will be encouraged to wear at meetings to show off their accomplishments\ shame other people into doing something. The documents referenced on this page are available at <http://www.min.uc.edu/robotics/research/tutorials>.

People will have a rank in each of the five following areas, starting at zero. The criteria for achieving higher levels are as follows:

**operator level 1:** can start robot, drive manually, put in line mode or nav mode, and qualify, given no other problems. can detect hardware problems. (read and understand Bearcat Micro Operator's Manual section 1-6; expected time: 1-2 hours)

**operator level 2:** can edit hardware.ini, write waypoint.ini, can read log files for demonstration purposes (read and understand Bearcat Micro operating instructions section 7 part 1-8; expected time: 4-8 hours)

**operator level 3:** can use robot operation and log files to diagnose issues (which algorithms or in hardware interface) and recommend fixes. (requires level 2 operator rank plus experience performing testing; expected time: indefinite)

**hardware interface programmer level 1:** can diagnose and fix\work around simple hardware problems. knows how to turn on and off individual components, can test individual components in different ways to debug. (read and understand lessons 1, 4, and 8; expected time: 2-3 hours)

**hardware interface programmer level 2:** can edit Robot\_Hardware library to work with a different component than is currently set up for. (attained after completing lessons 1-9 and assignments; expected time: a few hours a week for a quarter)

**hardware interface programmer level 3:** can edit Robot\_Hardware to add a completely novel component. (attained after working on lesson 10 group activity plus independent work; expected time: 10-20 hours)

**algorithm programmer level 1:** can produce a program that calls Robot\_Hardware, understands what algorithms are necessary to run competition, and given an algorithm, can interface it with Robot\_Hardware. (read and understand lesson 11, may need to reference previous lessons, i.e. lesson 1&6; expected time: 1-2 hours)

**algorithm programmer level 2:** understands basic functionality of 1 or more algorithms and can make small tweaks (see lesson 12-15 (target date for completion 2109; expected time: 20 hours)

**algorithm programmer level 3:** can read about or invent an algorithm and implement it in hardware. (get to level 2 algorithm programmer then go to grad school in robotics; expected time: relative)

**mechanic level 1:** can work with 80/20, rewire components

**mechanic level 2:** can solder, reassemble robot from scratch

**mechanic level 3:** can design and assemble circuit boards or build new robot from scratch

**secretary/treasurer level 1:** can complete and turn in basic paperwork without help, can order products without help.

**secretary/treasurer level 1:** can complete most paperwork, and

**secretary/treasurer level 3:** can complete and turn in a good funding report, or handle any other necessary paperwork.