

/*

001-Generated C code for functional specification 'COMPUTE_RUN_TIME'

VERSION: 3.2.3.9 C-RAT

: 1.6.0.0

AUTHOR: Hamilton Technologies Inc. Copyright 1991-2015.

OPERATION: COMPUTE_RUN_TIME

GENERATED: Tue Aug 25 16:55:18 2015

SCCSID: @(#) %M% %I% of %G%.

*/

#include "CAPABILITY.h"

#include "BOOLEAN.h"

#include "NAT.h"

#include "STR.h"

#include "CAPABILITIES.h"

#include "JOBOPERATION.h"

#include "JOBPROCESS.h"

#include "MACHINE.h"

#include "JOB.h"

#include "CHAR.h"

#include <stdio.h>

#include <math.h>

#include <errno.h>

#include "BOOLEAN.h"

#include "NAT.h"

fCOMPUTE_RUN_TIME(V0JOB, V0MACHINE,
 VORUNTIME)

IDECLARE_JOB(V0JOB)

IDECLARE_MACHINE(V0MACHINE)

ODECLARE_NAT(VORUNTIME)

{

```

/* __LOCAL_VARIABLE_DECLARATIONS__ */
DECLARE_NAT(C1)
DECLARE_NAT(V0TIMEPERUNIT)
DECLARE_NAT(V0UNITS)
DECLARE_STR(V0CAPANM)
DECLARE_BOOLEAN(V0HASOP)
DECLARE_CAPABILITY(V0C)
DECLARE_CAPABILITIES(V0CS)
DECLARE_CAPABILITY(V0CAPA)
DECLARE_STR(V0NM)
DECLARE_JOBOPERATION(V0OP)
DECLARE_JOBPROCESS(V0P)
DECLARE_BOOLEAN(D0D11)
DECLARE_CAPABILITIES(V1_0SN)
DECLARE_CHAR(C17)
DECLARE_BOOLEAN(V1_0ATN)
DECLARE_CAPABILITIES(V1_0S1)
DECLARE_NAT(C20)
DECLARE_BOOLEAN(V1_0ORNE)
DECLARE_BOOLEAN(V1_0ISR)
DECLARE_BOOLEAN(V1_0ISN)
DECLARE_BOOLEAN(V1_0ISE)
DECLARE_BOOLEAN(V1_0B0)
DECLARE_BOOLEAN(D1_0D8)
DECLARE_BOOLEAN(D1_0D13)
DECLARE_BOOLEAN(D1_0D17)
/* __ITERATION_VARIABLE_DECLARATIONS__ */
int rec17UNTIL;
DECLARE_CAPABILITIES(R171_0S1)
DECLARE_STR(R170NM)
/* __CONSTANT_DECLARATIONS_AND_ASSIGNMENTS__ */
NEWSTACK_TRACE_IDSC("COMPUTE_RUN_TIME");
DOT_K_NAT(0,C1)
DOT_K_CHAR('r',C17)
DOT_K_NAT(1,C20)
/* __FUNCTION_SOURCE_CODE_BEGINNING__ */

```

```

MOVETO_JOBPROCESS_JOB(V0JOB,V0P)
MOVETO_JOBOPERATION_JOBPROCESS(V0P,V0OP)
MOVETO_NAME_JOBOPERATION(V0OP,V0NM)
MOVETO_CAPABILITIES_MACHINE(V0MACHINE,V0CS)
ISEMPTY_CAPABILITIES(V0CS,V1_0ISE)
ISNULL_CAPABILITIES(V0CS,V1_0ISN)
ISREJECT_CAPABILITIES(V0CS,V1_0ISR)
OR_BOOLEAN(V1_0ISN,V1_0ISE,V1_0ORNE)
OR_BOOLEAN(V1_0ISR,V1_0ORNE,V1_0B0)
CLONE_BOOLEAN(V1_0B0,D1_0D8)
    if(D1_0D8<1)
{if(D1_0D8 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
    LOCATE_CAPABILITIES(C20,V0CS,V1_0S1)
R171_0S1=V1_0S1;
R170NM=V0NM;
rec17UNTIL=1;
while(rec17UNTIL--){
    ATNULL_CAPABILITIES(V1_0S1,V1_0ATN)
    CLONE_BOOLEAN(V1_0ATN,D1_0D13)
        if(D1_0D13<1)
{if(D1_0D13 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
    MOVETO_CAPABILITIES(V1_0S1,V0C)
    MOVETO_NAME_CAPABILITY(V0C,V0CAPANM)
    NOCASEEQ_STR(V0NM,V0CAPANM,V0HASOP)
    CLONE_BOOLEAN(V0HASOP,D1_0D17)
        if(D1_0D17<1)
{if(D1_0D17 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
    NEXT_CAPABILITIES(C17,V1_0S1,V1_0SN)
        rec17UNTIL=1;
        V1_0S1=V1_0SN;
        V0NM=V0NM;
}/*FALSE*/
else{/*CLONE1_CAPABILITY*/
    CLONE_CAPABILITY(V0C,V0CAPA)
}/*TRUE*/
}/*FALSE*/

```

```

else{ /*KREJECT_CAPABILITY*/
    KREJECT_CAPABILITY(V1_0S1,V0CAPA)
} /*TRUE*/
}
V1_0S1=R171_0S1;
V0NM=R170NM;
} /*FALSE*/
else{ /*KREJECT_CAPABILITY*/
    KREJECT_CAPABILITY(V0CS,V0CAPA)
} /*TRUE*/
    ISREJECT_CAPABILITY(V0CAPA,D0D11)
        if(D0D11<1)
            {if(D0D11 == REJECT_BOOLEAN) {REJECT_TEST_BOOLEAN()}
                MOVETO_UNITS_JOBOPERATION(V0OP,V0UNITS)
                MOVETO_TIME_CAPABILITY(V0CAPA,V0TIMEPERUNIT)
                MUL_NAT(V0UNITS,V0TIMEPERUNIT,*V0RUNTIME)
            }
} /*FALSE*/
else{ /*K_NAT*/
    K_NAT(C1,V0OP,*V0RUNTIME)
} /*TRUE*/
ENDSTACK_IDSC(); /* STR Garbage Collector */

return;
}
/* ----- end of source -----*/

```